

Fachhochschule Köln
University of Applied Sciences Cologne
Campus Gummersbach
Fakultät für Informatik und Ingenieurwesen

Fachhochschule Dortmund
Fachbereich Informatik

Verbundstudiengang Wirtschaftsinformatik

Diplomarbeit
(Drei-Monats-Arbeit)

zur Erlangung
des Diplomgrades
Diplom-Informatiker (FH)
in der Fachrichtung Informatik

„IMMOMENT,
eine datenbankgestützte Web-Applikation als
Informationsdienst für die Immobilienwirtschaft“

Erstprüfer: Prof. Dr. Heide Faeskorn-Woyke

Zweitprüfer: Prof. Dr. Andreas Liening

vorgelegt am: 14.06.2003

von cand. Michael Drzasga

aus Neumarktstr. 24
58095 Hagen

Tel.: 02 33 1 / 33 03 27

eMail.: michael.drzasga@gmx.de

Matr.-Nr.: 703 27 86

Unterschrift:

Inhaltsverzeichnis

VERZEICHNISSE	V
Abbildungsverzeichnis:.....	V
Tabellenverzeichnis:.....	V
Abkürzungsverzeichnis (alphabetisch sortiert).....	VI
ABSTRAKT	1
1 ASPEKTE DER SOFTWAREERSTELLUNG.....	8
1.1 Softwaretechnik	9
1.1.1 QUALITÄTSMANAGEMENT (QM)	10
1.1.2 VORGEHENSMODELLE.....	11
1.2 Die Art der Datenhaltung	14
1.3 Software-Ergonomie	16
1.3.1 GRUNDSÄTZE DER DIALOGGESTALTUNG	17
1.3.2 ANFORDERUNGEN AN DIE GEBRAUCHSTAUGLICHKEIT	17
1.4 Trennung von Daten und GUI.....	19
1.5 Arten von Internetseiten.....	20
1.6 Sicherheit von Internetseiten	24
2 TECHNIKEN.....	26
2.1 Die Skript-Sprache.....	27
2.1.1 AUFGABEN DER STEUERUNGSTABELLEN	28
2.1.1.1 Das Attribut „ID“	28
2.1.1.2 Das Attribut „zeile“	28
2.1.2 THEORETISCHE GRUNDLAGEN	28
2.1.3 ERKENNEN VON WERTEN („##“)	29
2.1.4 ERKENNEN VON BEFEHLEN („#?“)	29
2.2 Aufteilung von Logik- und Präsentationsschicht.....	31
2.3 Gauß-Krüger-Koordinaten.....	33
2.4 Datensuche durch Lage-Koordinaten.....	34
2.5 Datenschutz.....	35
3 DIE ENTWURFSPHASE	36
4 PARTNER.....	40
5 PRINZIP DES DATENAUSTAUSCHS	41
6 DATEN- UND KOMMUNIKATIONSMODELL.....	43

7	DATENBANKEN, DIE BASIS DES SYSTEMS	47
7.1	Die Datenbank im_basis.....	48
7.1.1	DIE NUTZERVERWALTUNGSTABELLE IM_USER_DATEN.....	49
7.1.2	IM_BUTTON_INFO.....	49
7.1.3	IM_GRUND_KOSTEN UND IM_KOSTEN_GRUPPEN	50
7.1.4	DAS BILLING-SYSTEM.....	51
7.1.4.1	<i>im_kosten_kopf (Kopf-Tabelle)</i>	51
7.1.4.2	<i>im_kosten_pos_fehler (Fehlersätze)</i>	52
7.1.4.3	<i>im_kosten_pos (Positionstabelle)</i>	52
7.1.5	KOMPETENZEN STEuern DIE DIENST-AUFRUFE.....	54
7.1.6	IM_GRUND_EINSTELLUNGEN.....	55
7.1.7	ADMINISTRATIONS-BEREICH.....	55
7.1.7.1	<i>im_user_erste_seite und im_status</i>	55
7.1.7.2	<i>im_se_admin</i>	55
7.1.8	IM_HILFE_TEXTE.....	56
7.1.9	IM_KOMMUNIKATION UND IM_KOMMUNIKATIONS_ART	56
7.1.10	PROTOKOLL-TABELLEN UND FEHLERMELDUNGEN	56
7.1.10.1	<i>db_protokoll, click_protokoll und im_zugriffe</i>	56
7.1.10.2	<i>im_error_messages</i>	56
7.1.11	IM_PARTNER	57
7.1.12	IM_BUDGET UND IM_BUDGET_USER.....	57
7.1.13	IM_FIRMEN_DATEN UND IM_FIRMEN_KOSTENSTELLEN	57
7.1.14	IM_KUNDE_SEIT.....	57
7.1.15	IM_ADRESSE	58
7.1.16	IM_DRUCK_BERECHTIGT	58
7.1.17	IM_KUNDEN_TELEINFO_PIC_LURA.....	58
7.2	Die Datenbank im_objekt.....	58
7.2.1	DARSTELLUNG DER WOHNUMFELD-BEZIEHUNGEN.....	59
7.2.2	IM_TELEINFO_STAEDTE.....	60
7.3	Die Datenbank im_leih.....	60
7.4	Die tempdb verwaltet die Session.....	62
8	WOHNUMFELDDATEN.....	67
8.1	Eine gekürzte Beispiel-Ausgabe	67
8.2	Steuerungstabelle zeilen_steuerung_wup	68
8.3	Die weiteren Tabellen.....	69
9	BRANCHEN IM UMFELD EINER IMMOBILIE.....	72
9.1	Der Dienst Infrastruktur	72
9.2	Der Dienst Branchenumfeld.....	75
10	DER DIENST OBJEKTLAGE	77
10.1	Stadt-, Lageplan, Topo-Karte und Luftbilder	79
10.2	City-View.....	82
10.3	City-Server.....	82
11	WEB-SERVICE AUFRUF	83
11.1	ImmoAgentAuthenticate Aufruf.....	83
11.2	ImmoAgentAuthenticate Rückgabe	84

12	SCHLUSSBEMERKUNG.....	85
12.1	Ziele.....	85
12.2	Ausblick.....	86
ANHANG A – PFLICHTENHEFT		87
1.	Zielbestimmung.....	87
1.1	MUSSKRITERIEN	87
1.2	WUNSCH- ODER KANN-KRITERIEN	88
1.3	ABGRENZUNGSKRITERIEN	88
2.	Produkt-Einsatz	89
2.1	ANWENDUNGSBEREICHE.....	89
2.2	ZIELGRUPPEN	89
2.3	BETRIEBSBEDINGUNGEN.....	89
3.	Produkt-Umgebung	90
3.1	SOFTWARE.....	90
3.2	HARDWARE.....	90
3.3	PRODUKTSCHNITTSTELLEN	91
4.	Produkt-Funktionalität	91
5.	Produkt-Daten	99
6.	Produkt-Leistungen.....	101
7.	Benutzeroberfläche	101
8.	Qualitätsbestimmung.....	102
9.	Globale Testfälle.....	102
10.	Entwicklungsumgebung	103
11.	Ergänzungen.....	103
ANHANG B – LITERATURVERZEICHNIS.....		104
ANHANG C – ERKLÄRUNG		105

Verzeichnisse

Abbildungsverzeichnis:

Abbildung 1 – Web-Ansicht des Hauptmenüs	5
Abbildung 2 – Web-Ansicht der Partner der IMMO-CHECK GmbH.....	40
Abbildung 3 – Prinzip des Datenaustauschs	41
Abbildung 4 – Daten- und Kommunikationsmodell	43
Abbildung 5 – Web-Ansicht – Adresseingabe: Nicht eindeutige Adresse.....	44
Abbildung 6 – Web-Ansicht Hauptmenü einer Straße mit unbekannter Hausnummer.....	46
Abbildung 7 – Datenbankschema der DB im_basis.....	48
Abbildung 8 – Datenbankschema der Wohnumfelddaten	59
Abbildung 9 – Web-Ansicht des Dienstes „Objektwert“	61
Abbildung 10 – Web-Ansicht des Dienstes Infrastruktur.....	72
Abbildung 11 – Web-Ansicht des Dienstes Branchenumfeld.....	75
Abbildung 12 – Web-Ansicht der Dienstes Objektlage	77
Abbildung 13 – Web-Ansicht des Dienstes Luftbilder	79
Abbildung 14 – Web-Ansicht des Dienstes City-Server.....	82

Tabellenverzeichnis:

Tabelle 1 – Grundsätze der Software-Ergonomie	16
Tabelle 2 – Inhalt der Tabelle im_button_info	49
Tabelle 3 – Inhalt der Tabelle im_grund_kosten.....	50
Tabelle 4 – Inhalt der Tabelle im_kosten_kopf.....	51
Tabelle 5 – Inhalt der Tabelle im_kosten_pos	52
Tabelle 6 – Inhalt der Tabelle im_kompetenz_gruppen	54
Tabelle 7 – Inhalt der Tabelle im_kompetenz	54
Tabelle 8 – Inhalt der Steuerungstabelle zeilen_steuerung_wup.....	68
Tabelle 9 – Inhalt der Tabelle zaehlweise_texte.....	69
Tabelle 10 – Inhalt der Tabelle s10_tabelle.....	69
Tabelle 11 – Inhalt der Tabelle Gemeinde.....	69
Tabelle 12 – Inhalt der Tabelle gemeinde_typ.....	69
Tabelle 13 – Inhalt der Tabelle gemeinde_groesse	69
Tabelle 14 – Inhalt der Tabelle STRAB_dom_Klassen.....	70
Tabelle 15 – Inhalt der Tabelle STRAB_BASIS	70
Tabelle 16 – Inhalt der Tabelle STRAB_RISIKO_STATUS_AUSLAND.....	70
Tabelle 17 – Inhalt der Tabelle STRAB_HAUSTYP.....	70
Tabelle 18 – Inhalt der Tabelle STRAB_MOSAICTYP_TOP3	70
Tabelle 19 – Inhalt der Tabelle kaufkraft_texte.....	70
Tabelle 20 – Inhalt der Tabelle Mosaic_gruppe.....	70
Tabelle 21 – Inhalt der Tabelle mosaic_typ.....	71
Tabelle 22 – Inhalt der Tabelle KREIS.....	71
Tabelle 23 – Inhalt der Tabelle h2_tabelle	71
Tabelle 24 – Inhalt der Tabelle h6_tabelle	71
Tabelle 25 – Inhalt der Tabelle HAUS_KERN	71
Tabelle 26 – Inhalt der Tabelle HAUS	71

Abkürzungsverzeichnis (alphabetisch sortiert)

.Net	spezielle Produktabkürzung	HTTP	Hypertext Transfer Protocol
3NF	3. Normalform	IIS	Internet Information Server
3-Tier	3-Schichten-Architektur	IL	Intermediate Language
ABC-Modell	Aufgabe-Benutzer-Computer-Modell	IP	Internet Protocol
Abk.	Abkürzung	ISAPI	Internet Server Application Programming Interface
AG	Aktiengesellschaft	ISO	Internationale Organisation für Standardisierung
ANSI	American National Standards Institut	IT	Informationstechnologie
API	application programming interface	ITU-T	International Telecommunications Union - Telecommunication Sector
ASCII	American Standard Code for Information Interchange	JIT	Just in Time
ASP	Active Server Pages	JPEG	Joint Photographic Experts Group
B2B	business to business	MIME	Multipurpose Internet Mail Extensions
BAFin	Bundesamt für Finanzdienstleistungsaufsicht	MS	Warenzeichen der Firma Microsoft
BDSG	Bundesdatenschutzgesetz	PDF	Portable Document Format
bzw.	beziehungsweise	PHP	Hypertext Preprocessor
BSI	Bundesamt für Sicherheit in der Informationstechnik	PKI	Public Key Infrastructure
C#	objektorientierte Programmiersprachen	PSID	Premier Support for Developer
C++			
CA	Certification Authority	QM	Qualitätsmanagement
CGI	Common Gateway Interface	QS	Qualitätssicherung
CLR	Common Language Runtime	s.	siehe (Verweis)
COM	Component Object Model	SOAP	Simple Object Access Protocol
CSS	Cascading Style Sheets	SQL	Structured Query Language
DB	Datenbank	SSL	Secure Socket Layer
DBMS	data base management system	SW	Software
DFSG	Debian Free Software Guidelines	TAG	XML oder HTML-Element
d.h.	das heißt	TCP	Transmission Control Protocol
DIN	Deutsches Institut für Normung	u.	und
DIN EN	vom DIN übernommene Europäische Normen	u.a.	unter anderem
dll	Dynamic Link Library	UDDI	Universal Description, Discovery and Integration
(E)DV	(elektronische) Datenverarbeitung	UNO	United Nations Organisationen
engl.	Englisch	URL	Uniform Resource Locator
ER-Modell	Entity-Relationship Modell (Diagramm)	VB	Visual Basic
EU	Europäische Union	WSDL	Web Services Description Language
EVA-Prinzip	Eingabe-Verarbeitung-Ausgabe – Grundprinzip der Datenverarbeitung	WWW	World Wide Web
FTP	File Transfer Protocol	WZ93	Wirtschaftszweige seit 1993
GIS	Geoinformationssystem	XML	Extensible Markup Language
GmbH	Gesellschaft mit beschränkter Haftung	XSL	eXtensible Stylesheet Language
GUI	graphical user interface	[x,S.y]	(In Fußnoten) der Verweis auf Nr. x des Literaturverzeichnisses mit S. (Seite) y
HTML	Hypertext Markup Language		

Abstrakt

Die vorliegende Diplomarbeit behandelt die Umsetzung der datenbankgestützten Web-Applikation IMMOMENT, mit dem Ziel primär anhand der Softwaretechnik¹ aufzuzeigen, in welchen Schritten IMMOMENT nach Anfertigung des Pflichtenheftes² umgesetzt wurde.

Kurzdarstellung IMMOMENT:

IMMOMENT wurde von der Firma IMMO-DATA AG³ (Bochum) für die Firma IMMO-CHECK GmbH (Bochum) entwickelt. Ziel war es Sachbearbeitern (z.B. einer Bank) die Möglichkeit zu verschaffen, über ein Immobilien-Objekt⁴ schnell und umfassend Informationen⁵ in Text und Bild zu bekommen.

¹ Prinzip: Modularisierung – Methode: Aufteilung des Gesamtproduktes in Teilprodukte, die nur über eine festgelegte Schnittstelle kommunizieren und sonst unabhängig sind. Definition von **Softwaretechnik** (software-engineering) aus „Lehrbuch der Software-Technik“ von H. Balzert: „Zielorientierte Bereitstellung und systematische Verwendung von Prinzipien, Methoden und Werkzeugen für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Softwaresystemen. Zielorientiert bedeutet die Berücksichtigung z.B. von Kosten, Zeit, Qualität.“[1]

² Das **Pflichtenheft** – (nach Balzert) ist das Ergebnisprotokoll der Definitionsphase (Teilschritt der Software-Entwicklung⁷). Das Pflichtenheft war Thema der Seminararbeit – siehe Anhang A

³ Kurzportrait der **IMMO-DATA AG** (einer Broschüre der Unternehmung entnommen):

„Die IMMO-DATA beschäftigt sich seit 1982 mit der Entwicklung von Beratungs- und Abwicklungssystemen für die Finanz- und Immobilienwirtschaft. Hierbei werden wesentliche Bestandteile aus der klassischen Backoffice-Anwendung in den mandantenfähigen Frontofficebereich zur Delegation an den POS (Point of Sale) verschoben, um Abwicklungsprozesse zu rationalisieren, Cross-Selling-Potenziale zu heben u. entsprechend die Effektivität u. Profitabilität der Kunden zu steigern. Seit über 20 Jahren entwickeln die IMMO-DATA zukunftsorientierte Softwarelösungen zu den Themen Finanzierungs- und Vermögensberatung inkl. Abwicklung, Bauträgercontrolling sowie Software rund um den Immobilienbereich. 1982 entstand das Produkt IMMOPLAN zur Bau- und Investitionsberatung, zu dem sich schnell weitere Lösungen zu finanzmathematischen Anwendungen und Bewertungssystemen addierten. 1995 setzte IMMO-DATA erstmals IMMOBANK (Bank- und Vermittlersystem zur (Direkt-) Zusage von Darlehensanträgen inkl. Scoring, Kreditvorlage, Genehmigung, fälschungssichere Vertragserstellung, etc. mit Übergabe Hostsystem) ein. 1997 wurden die Anwendungen webfähig. 2000 entstand zur Vermögensberatung das Produkt Lifeplan und zur Steuerung ganzer Portfolien IMMOPORTFOLIO. Durch die Verbindung der Produkte mit den von IMMO-CHECK generierten Informationen in dem von der IMMO-DATA entwickelten Produkt IMMOMENT besitzt die IMMO-DATA ein hervorragendes Instrumentarium zur Bewertung von Risiken im Hinblick auf die Anforderungen von Basel II und MaK.“

⁴ Ein **Immobilien-Objekt** (Hausanschrift) ist definiert durch eine eindeutige postalische Anschrift, der mindestens eine private oder gewerbliche Adresse zugeordnet werden kann.

⁵ „**Information** kann als jedes Zeichen oder Signal definiert werden, welche eine Organisation für die Steuerung und Regelung ihrer Tätigkeiten verwenden kann. Die Information unterscheidet sich durch die Zweckorientierung von einer Nachricht oder einem Signal. Informationen unterliegen keiner festen Struktur, wie sie aus einer Datei oder Tabelle bekannt sind.“ [7, S.12]
Information ist eine zweckbezogene Nachricht, die an einen Empfänger gerichtet ist.

IMMOMENT wurde aufgrund der Nachfrage verschiedener Großkunden und der Anforderungen gemäß Basel II⁶ entwickelt. Diese Kunden benötigten ein Produkt, das ihnen aktuelle Informationen über möglichst alle Immobilien-Objekte Deutschlands schnell zur Verfügung stellt.

Voraussetzung für das Projekt war, dass die zur Verfügung gestellten Daten einen bestimmten Qualitätsstatus aufweisen, der für den Kunden im besten Fall eine Vorort-Besichtigung nicht mehr nötig macht und damit eine deutliche Zeit- und Kostenersparung erbringt.

Eine einfache Marktanalyse in der Planungsphase⁷ ergab, dass keine andere Firma ein Produkt anbieten konnte, das die geforderten Qualitäts-Maßstäbe erfüllt. Neben der Marktanalyse sprach für ein solches Produkt, dass im Kreditgeschäft die Vorgaben von Basel II und MaK⁸ detailliertere Bonitätsprüfungen als bisher durch eigene interne Ratings⁹ der Kreditnehmer und des beliebigen Objektes (z.B. einer Immobilie) fordern.

⁶ „Die Neue Baseler Eigenkapitalvereinbarung, kurz **Basel II**, ist ein Regelwerk, das umfangreiche Vorschriften zur Eigenkapitalausstattung von Kreditinstituten beinhaltet. Basel II basiert auf den derzeit gültigen Regelungen, die im Jahr 1988 durch den Basler Ausschuss für Bankenaufsicht formuliert wurden (Basel I). Demzufolge müssen Kreditinstitute zur Absicherung von Risiken, insbesondere aus dem Kreditgeschäft, einen ausreichenden Betrag an Eigenkapital vorhalten. Auf diese Weise soll für den Fall Vorsorge getroffen werden, dass ein Kreditnehmer ausfällt, seinen Kredit also nicht zurückzahlt. Bisher wird für jeden Kreditnehmer der selbe Betrag an Eigenkapital vorgehalten, unabhängig davon, ob seine Bonität gut (tatsächliches Ausfallrisiko gering) oder schlecht (tatsächliches Ausfallrisiko hoch) ist. Diese Vorgehensweise ist jedoch nicht mehr zeitgerecht, denn die individuellen Risikoverhältnisse werden außer Acht gelassen. Deshalb wird der Eigenkapitalbedarf künftig am Risikogehalt der einzelnen Bankgeschäfte gemessen, um mehr Sicherheit und Stabilität im Kreditgewerbe zu gewährleisten. Denn die Zahlungsunfähigkeit eines Kreditinstitutes hätte fatale Folgen für die gesamte Bankenlandschaft und die Wirtschaft. In Deutschland werden die Regelungen zukünftig auf der Grundlage einer EU-Richtlinie in geltendes Recht umgesetzt.“ [10]

⁷ Die Software-Entwicklung wird (nach Balzert vergleiche [1]) in sechs Phasen aufgeteilt:

- **Planungsphase** (Kunde stellt Produzenten Produktwunsch vor. Der Wirtschaftlichkeitsanalyse folgt häufig ein Lastenheft, Systemfunktionen und Daten werden in groben Zügen festgelegt.)
- Definitionsphase (Systemanalyse - Ergebnisdokument ist das Pflichtenheft.)
- Entwurfsphase (Mit Mitteln der *Softwaretechnik* wird das Ergebnis der Definitionsphase genau beschrieben und zur Implementierung vorbereitet.)
- Implementierungsphase (Programmierung der Anwendung)
- Abnahme-/Einführungsphase (Das fertige System wird auf den Ernstfall vorbereitet und gestartet, oft müssen noch Daten aus Altsystemen übernommen werden.)
- Wartungs-/Pflephase (Zusätzlich nötige Erweiterungen werden eingebaut, bzw. gefundene Fehler, die sich erst bei der Laufzeit des Systems herausstellen, werden entfernt.)

⁸ Das Bundesamt für Finanzdienstleistungsaufsicht (BAFin) veröffentlicht die Mindestanforderungen an das Kreditgeschäft (**MaK**).

⁹ „Ein **Rating** ist ein Instrument zur systematischen Zuordnung von Kreditnehmern zu einer Ratingklasse [...]. Bei der Ermittlung dieser Ratingklasse werden verschiedene Faktoren betrachtet, die in ihrer Gesamtheit ein komplexes Bild von der Bonität des Kunden zeichnen. U.a. werden folgende Faktoren berücksichtigt: Bilanzdaten, betriebswirtschaftliche Steuerung und Planung, Kontoinformationen, Unternehmensführung, Rechtsform und Gesellschaftsstruktur. Die ermittelte Ratingklasse gibt Anhaltspunkte für die Fähigkeit eines Kreditnehmers, Kredit und die Zinsen zurückzahlen zu können.“ [10]

Basel II verlangt zusätzlich eine ausführliche und regelmäßige Prüfung zutreffender Risiken und deren zukünftiger Entwicklung durch ein einstellbares Rating, nachweis- und offenlegbar.

Um die gewünschten Leistungen anbieten zu können wurden kompetente Partner gefunden, die sehr viele Daten in Form von Text und Bild liefern (s. „Abbildung 1“ und nachfolgende kurze Darstellung der Dienste). Die Interessen der Partnern und Kunden beeinflussten, in welcher Form IMMOMENT entstand.

Es lag nicht im Interesse der Partner, dass deren Daten direkt beim Client vorliegen, aber auch die Datenmenge selbst stellte sich als zu groß für aktuelle Client-Systeme heraus, somit schied ein eigenständiges Client-Programm aus. Insbesondere in der ersten Phase, aber auch später sollten weitere Partner hinzukommen. Das führte dazu, dass die Variante eines „Fat Client“¹⁰, der über das Internet „nur“ noch die Daten von den Partner-Servern lädt, verworfen wurde, da häufige Updates bei den Kunden nötig gewesen wären.

Bei dieser Entscheidung musste auch den Interessen der Hauptkunden aus der Bankenbranche Rechnung getragen werden, die aus sicherheitstechnischen Gründen das Herunterladen von Daten aus dem Internet auf Client-Rechner nur mit strengen Auflagen zulassen.

Aus diesen Gründen wurde IMMOMENT als Web-Applikation¹¹ umgesetzt, wodurch ein schlanker Frontend realisiert werden konnte, somit ist clientseitig nur ein üblicher Internet-Browser¹² erforderlich.

¹⁰ Falls auf einem Client neben dem Abruf von Daten noch Berechnungen durchgeführt werden, so wird von einem **Fat Client** gesprochen.

¹¹ Bei einer **Web-Applikation** handelt es sich um eine Client-Server Architektur. „Unter der Client-Server-Architektur (engl.: client-server architecture) versteht man eine kooperative Informationsverarbeitung, bei der die Aufgaben zwischen Programmen auf verbundenen Rechnern aufgeteilt werden. In einem solchen Verbundsystem können Rechner aller Art zusammenarbeiten. Server (= Dienstleister; Backend) bieten über das Netz Dienstleistungen an, Clients (= Kunden; Frontend) fordern diese bei Bedarf an. Die Kommunikation zwischen einem Client-Programm und dem Server-Programm basiert auf Transaktionen, die vom Client generiert und dem Server zur Verarbeitung überstellt werden. Grundidee der Client-Server-Architektur ist eine optimale Ausnutzung der Ressourcen der beteiligten Systeme und ist das derzeit vorherrschende Konzept in der Datenverarbeitung von Rechnern, die über ein Kommunikationsnetz miteinander verbunden sind.“ [8]

¹² Ein **Browser** ist das Programm, mit dem man sich Internet-Seiten visualisieren kann.

Ein erster Prototyp¹³ mit noch wenigen Partnern wies manche Probleme auf.

Einige Partner mussten erst selbst Applikationen entwickeln, die IMMOMENT die benötigten Daten per XML¹⁴-Schnittstellen durch HTTP¹⁵-Requests¹⁶ oder über Web-Services¹⁷ übermitteln konnten.

Andere Partner stellte die Erstellung einer eigenen Web-Architektur vor zu große Probleme, diese übergaben der IMMO-CHECK ihre Daten komplett. Das hatte zur Folge, dass ein Volumen von mehreren Gigabyte Daten verwaltet werden musste. Hinzu kam das Problem, dass die gelieferten Daten in einer Form vorlagen, die möglichst wenig geändert werden sollte da sonst bei jedem Update Anpassungen nötig sein würden.

Neben dem Problem, auf welchem Wege und in welcher Form IMMOMENT an die Partnerdaten gelangt, war auch die Art der Abrechnung zu klären. Die Dienstaufrufe mussten mit den Partnern genau kalkuliert werden, da für die Web-Applikation weder ein Kaufpreis festgelegt werden konnte, noch ein Preis allein für die Nutzungsdauer in Frage kam. Des Weiteren musste bedacht werden, dass die einzelnen Dienstaufrufe später nachvollziehbar sein müssen, damit sowohl dem Nutzer eine Rechnung gestellt, als auch dem Partner die Kosten erstattet werden können.

¹³ „Ein **Prototyp** (engl.: prototype) ist ein ausführbares Modell (das heißt eine Vorversion) eines Informationssystem, wobei bestimmte Aspekte desselben hervorgehoben werden.“ [8]

¹⁴ „**XML** steht für „Extensible Markup Language“, in Deutsch „erweiterbare Auszeichnungssprache“. Sie wurde entworfen, um die Funktionalität des Web zu erweitern, indem man eine flexiblere und anpassungsfähige Identifizierung von Informationen ermöglicht.“ [12] „XML ist ein selbsterklärendes Dateiformat mit eigener Struktur. Somit ist es hervorragend geeignet um Textinformationen im Internet auszutauschen u. weiterzuverarbeiten. Eine XML-Struktur beinhaltet semistrukturale Daten, Bei semistrukturale Daten handelt es sich um Datenelemente, die entweder nur in Teilen strukturiert oder deren Struktur teilweise unregelmäßig ist. Die Struktur lässt sich nicht genau vor der Speicherung angeben, die Beschreibung der Struktur wird erst mit dem Datenelement selbst geliefert.“ [2, S. 874]

¹⁵ **HTTP** (Hypertext Transfer Protocol) technische Grundlage und Übertragungsprotokoll des WWW. Es ermöglicht den Zugriff auf Dokumente an Zielorten irgendwo im Netz (Gesamtheit der Internetseiten und Links, die gemeinsame Protokolle zum Austausch von Nachrichten verwenden). WWW bedeutet World Wide Web – „im WWW sind HTML-Dokumente durch URLs miteinander verknüpft. Dadurch realisiert das WWW einen Hypertext rund um die Welt. Das WWW bietet nicht nur Text, sondern über MIME-Erweiterungen auch Bild-, Ton- und andere Informationen. Das WWW ist mit Abstand der beliebteste Service im Internet geworden. Viele Leute setzen daher das Web mit dem Internet gleich.“ [14]

¹⁶ Im **HTTP-Request** (Client-Anfrage an den Web-Server) gibt der Client an, welche Protokollversion er verwendet, welches Dokument er spezifiziert und mit welcher Methode er es behandeln möchte.

¹⁷ „**Web-Services** stellen unabhängige, selbsterklärende und modulare Applikationen dar, die im Internet publiziert, lokalisiert und aufgerufen werden können. Web Services führen Funktionen aus - von simplen Anfragen bis zu komplizierten Geschäftsprozessen. Ist ein Web Service implementiert, können andere Applikationen (auch andere Web Services) diesen erkennen und aufrufen. Web Services nutzen verschiedene Standard-Protokolle, darunter: Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP) sowie Universal Description, Discovery and Integration (UDDI).“ [15] UDDI funktionieren im Internet wie „gelbe Seiten“. SOAP ist ein XML-basiertes Protokoll, ein einfacher Mechanismus zum Austausch strukturierter und typisierter Information zwischen den Rechnern in einer verteilten, dezentralisierten Umgebung.

Diese Form der Abrechnung (auf Basis „pay-per-Click“ bzw. „pay-on-Demand“) nach effektivem Use¹⁸ stellte für manche Partner einen neuen Weg dar, so dass diese ihre Systeme erst auf diese Abrechnungsweise anpassen mussten (s. „Abbildung 1“).

Es waren aber nicht nur Probleme mit den Partnern zu bewältigen, denn die Art der Umsetzung als Web-Applikation brachte auch einige Nachteile mit sich:

- Sicherheitsrisiken,
- die Geschwindigkeit der Übertragung ist abhängig von der Netzbelastung und der zu übertragenden Datenmengen,
- es muss serverseitig ermittelt werden können, ob die Daten im Client-Browser angezeigt worden sind,
- aber auch gestalterische bzw. ergonomische Nachteile durch die eingeschränkte Benutzeroberfläche mussten berücksichtigt werden.

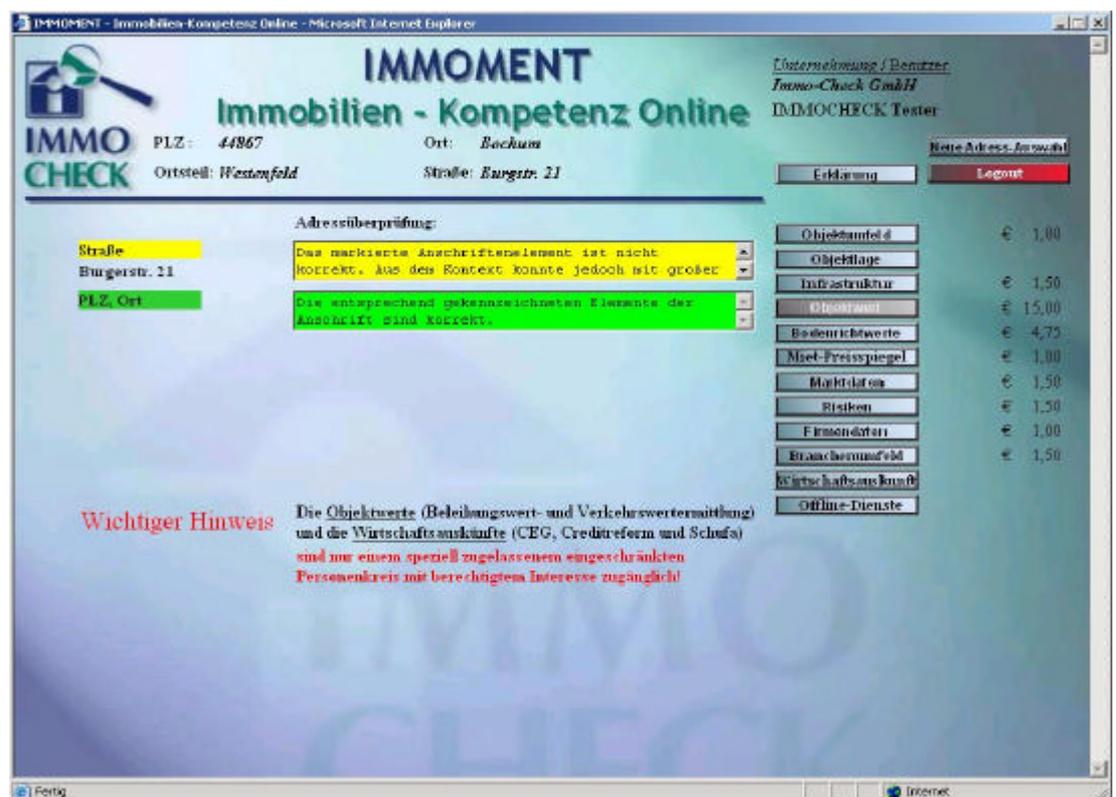


Abbildung 1 – Web-Ansicht des Hauptmenüs

In der „Abbildung 1“ sind auf der rechten Seite die Kosten zu sehen, die der Nutzer zahlen muss, wenn der entsprechende Dienst (Button links daneben) angewählt wurde. Neben der Kostenübersicht zeigt das Hauptmenü in „Abbildung 1“ die Dienste, die IMMOMENT anbietet, nachfolgend eine kurze Darstellung der Dienste, im „Anhang A – 4. Produkt-Funktionalität“ werden alle Dienste detailliert beschrieben:

¹⁸ Use (engl. Nutzung) – in diesem Falle ist der Aufruf einzelner IMMOMENT-Dienste gemeint.

- **Objektumfeld (s. Kapitel 8 Wohnumfeld)**
Zur Beurteilung der Immobilie, seiner Nutzung, Lage und seines Umfeldes werden in Textform Informationen geliefert, die auf einer speziellen Auswertung einer mikrogeographischen Datenbank eines Partners basieren.
- **Objektlage (s. Kapitel 10)**
Dieser Dienst liefert Bildinformationen.
- **Infrastruktur (s. Kapitel 9)**
Der Dienst zeigt spezielle Branchen, wie Ärzte, Schulen oder Kirchen, die in einem bestimmten Umkreis um das Immobilien-Objekt vorhanden sind.
- **Objektwert (s. Kapitel 7.3)**
Falls vorhanden werden Beleihungswertermittlungen zur Immobilie angezeigt, die ausschließlich von berechtigten Personen erfasst wurden.
- **Bodenrichtwerte**
Bodenrichtwerte sind Orientierungswerte zur Einschätzung des Marktwertes von Grundstücksflächen – ohne deren Gebäudewert.
- **Miet- Preisspiegel**
Ausgegeben werden Mietspiegel für Wohnobjekte, Kaufpreise für Eigentumswohnungen, Reihenhäuser / Doppelhaushälften und Ein- Zwei-Familienhäuser.
- **Marktdaten**
Es wird die Marktpreis-Entwicklung bestimmter Werte (z.B. Mieten für Eigentumswohnung, Neubau, oder Altbau) einer Gemeinde seit 1975 angezeigt.
- **Risiken**
Der Dienst gibt Werte spezieller Indikatoren (z.B. Zahlungsrisiko) aus, die wichtige Hinweise über die mit dem Objekt verbundenen Risiken liefern.
- **Firmendaten**
Neben der Branche wird für vorhandene Firmen der ungefähren Umsatz des Vorjahres und die Mitarbeiterzahl angegeben.
- **Branchenumfeld (s. Kapitel 9)**
Der Dienst zeigt durch Vorgabe eines Umkreises um das Objekt herum und nach Auswahl verfügbarer Branchen an, welche Firmen zu der Branche vorhanden sind.
- **Wirtschaftsauskunft**
Dieser Dienst muss noch umgesetzt werden (s. Schlussbemerkung – Ausblick)
- **Offline-Dienste**
Der Dienst veranlasst per eMail, dass ein Partner vor Ort eine Beschreibung der Immobilie in Text und / oder Bild erstellt, bzw. einen Grundbuchauszug besorgt.

Es folgt eine Kapitelübersicht. Die weiteren Kapitel behandeln hauptsächlich zwei Aspekte: Erstens, wie die beschriebenen Aufgaben gelöst wurden, und zweitens, welche Informationen in Text und Bild IMMOMENT seinen Kunden liefert.

Die ersten beiden Kapitel „Aspekte der Softwareerstellung“ und „Techniken“ beschäftigen sich mit der Beantwortung der ersten Frage. Im ersten Kapitel werden bestimmte Punkte der Softwaretechnik besonders hervorgehoben, zusätzlich beinhaltet es ergonomische¹⁹ Aspekte und die speziell bei Web-Applikationen zu berücksichtigenden Punkte: Arten und Sicherheit von Web-Seiten.

Das zweite Kapitel behandelte eingesetzte Techniken: Die Skript-Sprache, die Entwicklungsumgebung und Bildanzeige bzw. Datensuche durch Lage-Koordinaten, aber auch das Thema „Datenschutz“, da insbesondere die Wohnumfelddaten aus Sicht des Datenschutzes kritische Informationen anzeigt.

Mit der Entwurfsphase zeigt das dritte Kapitel die noch speziell zu berücksichtigenden Punkte bei der Umsetzung von IMMOMENT auf.

Um IMMOMENT umsetzen zu können waren Partner nötig, die im vierten Kapitel mit deren eingebrachten Daten kurz vorgestellt werden.

In den beiden nachfolgenden Kapiteln wird aufgezeigt, wie der Datenaustausch und die Kommunikation zwischen den Partnern und IMMOMENT funktionieren. IMMOMENT „lebt“ durch die Datenvielfalt, die speziell im Kapitel sieben „Datenbanken, die Basis des Systems“ dargestellt wird.

In den Kapiteln acht bis elf werden die Dienste von IMMOMENT vorgestellt, die seit Erstellung des Pflichtenheftes (s. Anhang A - 4. Produkt-Funktionalität) hinzugekommen sind bzw. sich verändert haben. Dabei zeigen die Kapitel neun und zehn, dass IMMOMENT zwar kein Geoinformationssystem (GIS²⁰) ist, aber die Lage-Koordinaten ermöglichen IMMOMENT die Darstellung von Daten und Bilder, die sich auf die Lage der Immobilien, bzw. auf deren Umfeld beziehen.

Im letzten Kapitel „Schlussbemerkung“ wird dargestellt, ob die anfangs gesteckten Ziele von und mit IMMOMENT erreicht wurden. Ein Ausblick auf mögliche bzw. bereits geplante Erweiterungen schließt die Arbeit ab.

¹⁹ „Die **Ergonomie** (engl.: ergonomics) ist die Lehre von der Arbeit des Menschen (in physiologischer und funktioneller Hinsicht) und seinen Beziehungen zur Arbeitsumgebung (benutzte Arbeitsmittel u. Maschinen). Software-Ergonomie hat zum Ziel Computersysteme zu entwickeln, die eine problemlos und intuitiv anwendbare Funktionalität besitzen, die einfach zu erlernen u. zu bedienen sind.“ [8]

²⁰ **GIS** (Geographisches Informationssystem) „Geoinformationssysteme sind Computersysteme zur Verarbeitung räumlicher Daten mit räumlichen und topologischen Eigenschaften. Die Daten haben eine an die Systeme angepasste Struktur, die geometrische Grundelemente wie Punkt, Linie und Fläche verarbeiten kann und geometrische Berechnungen erlaubt.“ [4, S. 17]

1 Aspekte der Softwareerstellung

In diesem Kapitel werden die Aspekte der Softwareerstellung aufgeführt, die insbesondere bei der Web-Applikation IMMOMNET zu beachten waren.

Wie bereits erwähnt, wurde IMMOMENT unter der Prämisse der stetigen „Erweiterbarkeit“ erstellt, da nach und nach weitere anzuzeigende Daten von neuen Partnern hinzukamen und auch später hinzukommen werden. Aus diesem Grunde wird das Vorgehensmodell behandelt und vorab in den Gesamtrahmen der Softwaretechnik eingeordnet.

IMMOMENT liefert dem Kunden eine große Anzahl von Textinformationen, die von einigen Partnern nicht über das Internet abgefragt, sondern auf einem eigenen Datenbank-Server verwaltet werden. Neben diesen Daten müssen weitere Verwaltungsdaten wie die Kostentabellen gespeichert werden, so war es nötig, in einem Unterkapitel „die Art der Datenhaltung“ vorzustellen.

Es müssen aber auch Einschränkungen durch die Bedienungsfläche in Kauf genommen werden, die im Unterkapitel „Software-Ergonomie“ behandelt werden.

In dem Unterkapitel „Trennung von Daten und GUI“ wird die 3-Tier-Architecture vorgestellt, die es ermöglichte, das Frontend möglichst einfach zu halten, das für IMMOMENT eine wichtige Voraussetzung darstellte.

Daran knüpft das Unterkapitel „Arten von Internetseiten“ an, in dem das Thema „Thin Client“ aus einer anderen Sicht betrachtet wird. Das Unterkapitel „Sicherheit von Internetseiten“ beendet die „Aspekte der Softwareerstellung“, dort werden die derzeit gebräuchlichsten Wege dargestellt Daten im Internet so zu transportieren, dass sie nicht von Dritten gelesen oder gar verfälscht werden können.

1.1 Softwaretechnik

„Die Kunst der Programmierung“ führte zu vielen Entwicklungsfehlern, die spätestens in den sechziger Jahren in der Softwarekrise einen Umdenkungsprozess auslösten. Man erkannte, dass schon aus Gründen der Übersichtlichkeit Software, wie andere Produkte auch, strukturiert und nachvollziehbar erstellt werden muss – die Softwareentwicklung und Softwaretechnik wurden ausgebildet.

Die Softwaretechnik befasst sich mit der professionellen Entwicklung von Anwendungssoftware²¹-Systemen. Die Aufgabe der Softwaretechnik ist nicht nur die Komplexität großer Anwendungssoftware in überschaubare Teile zu zerlegen, sondern auch Qualitätsanforderungen zu erfüllen.

Die Qualitäts²²-Anforderungen bestehen aus den zwei Bereichen Brauch- und Wartbarkeit, die in der DIN 9126 wie folgt aufgeteilt werden: Funktionalität, Zuverlässigkeit, Benutzbarkeit, Sicherheit, Korrektheit, Robustheit, Ergonomie, Effizienz, Testbarkeit, Überprüfbarkeit, Erweiterbarkeit und Wiederverwendbarkeit.

²¹ „Die **Anwendungsprogramme** (engl.: application program; user program) bieten im Gegensatz zu Systemprogrammen Lösungen für fachliche Probleme. Dazu gehören technisch-wissenschaftliche Programme (z.B. für statistische Berechnungen), kommerzielle, auf allgemeine betriebliche Funktionen bezogene Programme (z.B. für die Finanzbuchhaltung) und Branchenprogramme.“ [8]

²² **Qualität** ist abgeleitet aus dem lateinischen Wort „qualitas“ : Beschaffenheit, Eigenschaft – Eignung eines Gutes für bestimmte Verwendungszwecke (Zweckeignung). Auf Software bezogen bedeutet es eher: Gesamtheit von Eigenschaften eines SW-Produktes, die sich auf die Eignung zur Erfüllung gegebener Erfordernisse beziehen. Die DIN 9126 formuliert es noch genauer: „Softwarequalität ist die Gesamtheit der Merkmale und Merkmalswerte eines SW-Produkts, die sich auf dessen Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse zu erfüllen.“ [16]

Neben diesen allgemeinen²³ werden möglicherweise spezielle Qualitäts-Anforderungen (Qualitätszielbestimmungen) im Pflichtenheft festgehalten, das kann bedeuten, dass dem Auftraggeber ein bestimmtes Qualitätsmerkmal sehr wichtig ist, welches höchste Priorität erlangt. Ein solch spezielles Merkmal könnte bei einer Web-Applikation sein, dass die Anwendung „rund um die Uhr“ genutzt werden kann.

1.1.1 Qualitätsmanagement²⁴ (QM)

QM-Maßnahmen gewährleisten durch die Qualitätssicherung²⁵, dass geprüft wurde, ob die gestellten Qualitätsanforderung erfüllt worden sind. Das Bereitstellen eines „passenden“ Vorgehensmodells²⁶ ist dabei die wichtigste Grundlage der konstruktiven QM-Maßnahme, aber auch die Einführung von Werkzeugen und die Schulung der Mitarbeiter müssen gegeben sein. An die konstruktiven QM-Maßnahmen schließen sich die analytischen an, die aber erst während der Entwicklung prüfen, ob die geforderten Regelungen eingehalten wurden.

²³ Die **allgemeinen Qualitätsanforderungen** lassen sich grob in zwei Gruppen unterteilen:
1. innere Qualität von SW für dem Entwickler wichtig (z.B. Korrektheit, Robustheit, Testbarkeit, Erweiterbarkeit und Wiederverwendbarkeit)

2. äußere Qualitätsmerkmale für den Nutzer wichtig (z.B. Brauchbarkeit, Ergonomie u. Effizienz).
²⁴ Unter **Qualitätsmanagement** (QM) versteht man Planung, Steuerung und Kontrolle aller Aufgaben, um die Qualität von Prozessen und Produkten sicherzustellen.

²⁵ „Unter **Qualitätssicherung** (quality assurance – Abk.: QS) versteht man die Summe aller Maßnahmen, die die Qualität des entstehenden Produktes gewährleisten sollen.“ [1]
Begriffsabgrenzung: QM ist der globale Begriff, er umfasst alle Maßnahmen zur Sicherstellung der Qualität in einem Betrieb. QS hingegen bezieht sich immer auf ein konkretes Projekt, also auf die Entwicklung eines konkreten IT-Systems. (IT -Abk. für Informationstechnik bzw. -technologie).

²⁶ „Ein **Vorgehensmodell** (life cycle models) dient zur Benennung und Ordnung von produktbezogenen Tätigkeiten bei der Softwareerstellung“. [2, S. 776] „Ein Vorgehensmodell ([...] Prozessmodell) ist ein Regelwerk zur Sicherstellung des geordneten Ablaufs einer Entwicklung.“ [3, S. 15] Es wird die Vorgehensweise modelliert, d.h. man überlegt, in welcher Reihenfolge ein typischer Softwareentwicklungsprozess abläuft, mit der Fragestellung: Gibt es bestimmte Stationen oder bedeutende Schritte, die erreicht oder ausgeführt werden sollten?

1.1.2 Vorgehensmodelle

Es gibt unterschiedliche Vorgehensmodelle, die sowohl von der Komplexität des zu erstellenden Softwareproduktes und den Qualitätszielen abhängen als auch den gegebenen Ressourcen (Finanzielle Mittel, Zeit und Mitarbeiter) Rechnung tragen.

Es werden mehrere grundsätzliche Typen von Vorgehensmodellen unterschieden:

- Phasen- oder Wasserfallmodell²⁷
- V-Modell²⁸
- Unterstützung nebenläufiger Aktivitäten²⁹
- Evolutionäres Vorgehen³⁰
- Inkrementelles Vorgehen³¹
- Prototyping³²

²⁷ „Das grundlegende Projektmodell zur Softwareproduktion ist das lineare **Phasen- oder Wasserfallmodell**, das die Herstellung von Software als Folge von Abschnitten (Phasen) beschreibt. Es existiert in zahlreichen Variationen und bietet benannte und standardisierte Entwicklungsschritte, die zeitlich sequentiell durchlaufen werden sollen und zu Ergebnissen in Form von resultierenden Dokumenten führen. Rückgriffe auf vorangegangene Phasen sind meist erlaubt und in der Praxis die Regel.“ [2, S. 776] Z.B.: Analyse und Definition – Entwurf – Implementation – Test – Einsatz und Wartung

²⁸ „**V-Modelle** berücksichtigen die Symmetrie der Phasen, die ersichtlich wird, wenn man die Zeitachse in der Mitte des Entwicklungsprozesses V-förmig verbiegt: Auf dem linken Schenkel des „V“ wird das System immer feiner spezifiziert, bis schließlich die Module als kleinste Systemteile implementiert werden können. Die Phasen auf dem linken Schenkel korrespondieren mit prüfenden oder integrierenden Phasen auf dem rechten Schenkel. Vorteil gegenüber dem Phasenmodell ist die Etablierung qualitätssichernder Maßnahmen. (B.W.Boehm von)“ [3, S. 19]

²⁹ „Bei streng klassischer Auslegung kann eine Phase der Systementwicklung erst dann in Angriff genommen werden, wenn die vorangegangene Phase in Gänze abgeschlossen ist. Zur Unterscheidung vom klassischen, streng sequentiellen Vorgehen wird im Zusammenhang mit Nebenläufigkeit der Terminus Aktivität statt Phase bevorzugt. Durch **Unterstützung nebenläufiger Aktivitäten** ist es nun möglich, dass freie Ressourcen mit parallelen Aktivitäten betraut werden, allerdings sind bestimmte Synchronisationspunkte zu beachten.“ [3, S. 20]

³⁰ „Phasen-, Wasserfall- und V-Modelle haben eines gemeinsam: Sie zielen darauf ab, das gewünschte IT-System in einem einzigen Durchgang durch ihre Phasen (bzw. bei nebenläufigem Ansatz: durch ihre Aktivitäten) zu erstellen, und das auch noch zur vollen Zufriedenheit des Auftraggebers. Dieser Ansatz wird auch treffend als „big bang“, „one shot“ oder „grand design“ bezeichnet. Ein anderes Extrem ist das **Evolutionäre Vorgehen**: Ausgehend von zunächst nur grob fixierten Anwenderforderungen wird ein erstes ‚V‘ durchlaufen und das erstellte System dem Anwender vorgestellt, der somit eine Grundlage hat, auf der er seine Vorstellungen konkretisieren kann. In einem weiten ‚V‘ wird das System den konkretisierten Anforderungen angepasst, usw.“ [3, S.21]

³¹ „Beim **Inkrementellen Vorgehen** wird ein realistischer Kompromiss zwischen „big bang“ u. evolutionärem Ansatz geschlossen. Die Anwenderforderungen werden zu Projektbeginn möglichst exakt spezifiziert. Wichtig ist hierbei eine saubere Gliederung des künftigen Systems in mehrere Teilsysteme, die zum Teil auch einzeln verwendbar sein sollen. Auf dieser Grundlage wird ein erstes Teilsystem in einem schmalen ‚V‘, dem das obere Ende des linken Schenkels fehlt, entwickelt u. beim Anwender in Betrieb genommen. In einem zweiten, ebenso reduzierten ‚V‘ wird ein zweites Teilsystem entwickelt, mit dem ersten integriert u. ebenfalls beim Anwender in Betrieb genommen. Der Vorgang wiederholt sich, bis schließlich das Gesamtsystem beim Anwender vorliegt.“ [3, S. 22]

³² „Ein **SW-Prototyp** wird vor der eigentlichen Systementwicklung (meist möglichst schnell) erstellt u. berücksichtigt nur einen ausgewählten Teil der Anforderungen, die an das endgültige System gestellt werden. Typisch für die Erstellung eines Prototyp ist die Beschränkung auf ausgewählte Anforderungen [(Klärung von Anwenderwünschen, Nachweis der Realisierbarkeit, Experimentieren). Man unterscheidet Prototypen nach den Anforderungen in Oberflächen- o. Funktionsprototyp.]“ [3, S. 23]

IMMOMENT wurde aus einer Kombination von „Evolutionärem Vorgehen“ und „Prototyping“ erstellt.

Nach der Marktanalyse in der Planungsphase wurde zum Nachweis der Realisierbarkeit ein Funktionsprototyp erstellt. Der Prototyp zeigte, dass mit den aktuell genutzten Technologien und den gegebenen Ressourcen die Erstellung einer ersten Version in wenigen Monaten kaum möglich sein würde. Die zu dem Zeitpunkt genutzten Technologien waren: XML, XSL³³ in HTML³⁴-Seiten, ASP-Seiten steuerten diesen Aufbau, berechnungsintensivere Logik wurden durch COM³⁵-Komponenten (die in C++ erstellt wurden) verrichtet.

Durch die insbesondere am Anfang des Projektes hinzukommenden Partner musste IMMOMENT häufig erweitert werden. Aus diesem Grunde wurde das „Evolutionäre Vorgehen“ angewendet, wodurch den Kunden und Partnern der neue Stand alsbald erneut vorgestellt werden konnte.

IMMOMENT stellt als „Informationssammler“ von Partner-Web-Servern eine besondere Art der Programmarchitektur dar, so dass sich sehr früh herausstellte, dass „alte“ Anwendungen, besser gesagt bereits vorhandene globale Strukturen und komplexe mathematische Berechnungsfunktionen nicht genutzt werden konnten. Aus diesem Grunde ergab sich die Möglichkeit IMMOMENT als ein komplett neues Produkt zu erstellen, da keine vorhandenen „Pfade“, wie Schnittstellen oder Architekturen, berücksichtigt werden mussten.

³³ **XSL** (eXtensible Stylesheet Language) ist eine Stylesheetsprache. Sie legt genau fest, wie die Daten (=XML) angezeigt werden. Ein Stylesheet ist ein wohlgeformtes XML-Dokument, das mit einer Reihe von speziellen Anweisungen die Daten in der XML-Datei umwandelt. Wenn auf eine XML-Datei ein Stylesheet angewendet wird, steuert das Stylesheet die Formatierung bzw. die Darstellung der XML-Datei.

³⁴ **HTML** steht für Hypertext Markup Language, sie ist eine einfache Beschreibungssprache für Dokumente, die plattformunabhängig arbeiten.

³⁵ **COM** (Component Object Model) definiert einen allgemeinen Weg für den Zugriff auf SW-Dienste.

Es wurde entschieden, IMMOMENT mit .Net³⁶-Framework³⁷ von Microsoft und der objektorientierten Programmiersprache C# komplett neu zu erstellen, insbesondere nachdem Schulungen einen deutlichen Zeitgewinn bei der Oberflächen-Erstellung mit .Net aufwiesen.

Zur Unterstützung wurde ein PSfD³⁸-Vertrag mit Microsoft abgeschlossen, der gewährleistet, dass ein Microsoft-Consultant³⁹ dieses Projekt unterstützt und nicht nur bei Problemen Hilfestellung leistet, sondern auch beratend bei Architektur-entscheidungen zur Seite steht.

³⁶ „Das Ziel von **.NET** besteht darin, eine Software-Technologie zur Verfügung zu stellen, die einen Informationsaustausch zwischen den unterschiedlichsten Geräten und Betriebssystemen erlaubt. Im Gegensatz zu Java soll dabei allerdings nicht eine Programmiersprache der Kitt sein, der das Ganze zusammenhält, sondern diese Aufgabe übernehmen die Web-Services. Hinter diesem neuen Marketing-Begriff steckt der Informationsaustausch zwischen beliebigen Systemen über Protokolle wie zum Beispiel XML und HTTP, die inzwischen zum allgemein akzeptierten Industrie-Standard geworden sind. Da Web-Services nicht nur von Microsoft, sondern von allen namhaften Soft- und Hardware-Herstellern unterstützt und weiterentwickelt werden, können Sie davon ausgehen, nicht aus Versehen in eine Sackgasse einzubiegen. Statt dessen schaffen die Web-Services eine funktionale Plattformunabhängigkeit und erlauben den Zugriff auf Server, unabhängig davon, welches Betriebssystem der Server gerade nutzt. Und im Gegensatz zu Java müssen Sie nicht die gewohnte Programmiersprache wechseln, sondern können bei der Muttersprache bleiben – solange diese .NET unterstützt.“ [9]

³⁷ „Das **.NET Framework** ist Teil von Microsofts .NET-Strategie. Es ist eine Softwareplattform, die Ersatz für das wild gewachsene Win32-API sein soll, allerdings wesentlich mehr bietet als eine Bibliothek mit Befehlen. Um in einer heterogenen Welt bestehen zu können, soll das .NET Framework Plattformunabhängigkeit bieten, eine einheitliche Programmierschnittstelle ähnlich der Java Runtime Environment. Allerdings geht MS etwas anders an die Sache heran.“ [9]

³⁸ Auszug der Erläuterung der Microsoft Homepage: „**PSfD** (Premier Support for Developer) bietet Ihnen nicht nur die Möglichkeit der Problembeseitigung. Der eigentliche Schwerpunkt liegt auf der proaktiven und intensiven Beratung Ihres Unternehmens bei der Entwicklung von Softwarelösungen. Mit PSfD stellen Sie sicher, dass Sie auf der Basis von und mit Microsoft-Produkten Softwarelösungen entwickeln, die die Möglichkeiten der Microsoft-Technologien optimal nutzen.“ [13]

³⁹ **Consultant** ist ein Berater, der für die verschiedensten Unternehmensbereiche Dienste verrichtet.

1.2 Die Art der Datenhaltung

Textinformationen zu allen Immobilien anzeigen zu können stellt die Hauptleistung von IMMOMENT dar. Viele dieser Daten werden von einem eigenen Datenbank-Server verwaltet, da es für einige Partner zu aufwendig war, diese Daten über das Internet zu liefern. Neben diesen Daten müssen weitere Verwaltungsdaten, wie die Kostentabellen gespeichert werden, das macht es nötig die Datenhaltung vorzustellen. Die Web-Applikation IMMOMENT wird hauptsächlich durch ein Datenbanksystem⁴⁰ gesteuert, das aus fünf Datenbanken⁴¹ besteht, die der SQL-Server verwaltet.

Aus Datensicht durchläuft der Datenbankentwurf einer relationalen Datenbank⁴² mehrere Schritte. Der Datenbankentwurf ist eine Verfeinerung des ANSI⁴³-3-Ebenen - Modells⁴⁴. Auf die konzeptionelle Ebene (konzeptionelles u. logisches Schema⁴⁵) folgt das Datenbankschema. Das normalisierte logische Schema erweitert um Views⁴⁶, Indizes⁴⁷ und Zugriffsrechte⁴⁸ stellt das Datenbankschema dar.

⁴⁰ „Das Wesentliche bei einem **Datenbanksystem** (engl.: data base system) ist die Ausübung einer zentralen Kontrolle über eine von vielen Benutzern beziehungsweise Programmen verwendete Datenmenge. Es wird ein globales Modell festgelegt, das den für den Betrieb interessanten Realitätsausschnitt widerspiegelt (= konzeptionelles Modell). Die Probleme der Datenspeicherung und -organisation werden zentral gelöst und von den speziellen Auswertungen der Daten für die verschiedenen fachlichen Aufgabenstellungen abgekoppelt.“ [8]

⁴¹ „Die Bestandteile einer **Datenbank** (engl.: data base; Abk.: DB), sind die eigentlichen Daten und das Datenbankverwaltungssystem (engl.: data base management system; Abk.: DBMS). Das DBMS ist ein Programmsystem zum Aufbau, zur Kontrolle und zur Manipulation der DB.“ [8]

⁴² „Ein Datenbankmodell stellt einen allgemeinen Begriffsapparat zur Verfügung, der es gestattet, Realitätsausschnitte ohne Eingrenzung auf ein bestimmtes Sachgebiet zu modellieren. Mit dem Datenmodell werden die Datenobjekte und die Operationen auf diesen Objekten festgelegt. Das Datenmodell ist somit analog zu einer Programmiersprache, die Datentypen und Funktionen werden mit den Mitteln der Datenbanksprache beschrieben.“ [4, S. 20] Im **Relationalen Datenbankmodell** (entwickelt 1970 von E. F. Codd) werden die Daten und Beziehungen in Tabellen (Relationen) abgebildet. Die Tabellen sind weitgehend unabhängig voneinander.

⁴³ **ANSI** (American National Standards Institut) bezeichnet das amerikanische Normungsinstitut.

⁴⁴ „Datenbankarchitektur aus Anwendungssicht, das **ANSI-3-Ebenen Modell** unterscheidet 3 Sichten:

- Die physische Sicht auf die Daten (interne Ebene vom Datenbankadministrator verwaltet)
Bestandteil: Informationen über die Art und den Aufbau der Datenstrukturen auf dem physikalischen Speicher und die Zugriffsmechanismen.
- Die logische Gesamtsicht der Daten (konzeptionelle Ebene - vom Datenbankentwickler verwaltet)
Bestandteil: Logische Darstellung der Gesamtsicht der Daten in einem speziellen Datenmodell, teilt sich auf in konzeptionelles (unabhängig vom Datenmodell) und logisches Schema (angepasst auf das Modell eines bestimmten Datenbanktyps z.B. auf das relationale DB-Schema)
- Die Benutzersicht auf die Daten – (externe Ebene - vom Datenbankentwickler verwaltet)“ [4,S.35]

⁴⁵ „Das **logische Schema** folgt dem konzeptionellen Schema (wird durch ER-Diagramme veranschaulicht). Es bereitet die Umsetzung für ein bestimmtes Datenbankmodell vor. Zur Erstellung des logischen Schemas wird das ER-Diagramm transformiert, zwei Hauptregeln sind zu beachten:

- Beziehungen werden auf Attribute (1:n) oder Relationen (n:m Beziehungen) abgebildet
- Die Entitäten des ER-Diagrammes werden auf Relationen abgebildet“ [Vergl. 4 LE S. 27ff]

⁴⁶ **Views** sind gespeicherte DB-Abfragen. Sie enthalten selbst keine Daten, stellen aber Informationen anderer Tabellen zur Verfügung. Auf sie kann wie auf „normale“ Tabellen zugegriffen werden.

⁴⁷ **Indizes** sind DB-Objekte, die für eine Tabelle erstellt werden können, um den direkten Zugriff auf bestimmte Datenzeilen (Attribute) zu beschleunigen.

Die ER-Diagramme⁴⁹ der Planungsphase wurden in logische Schemata transformiert, die Datenbanken von IMMOMENT wurden normalisiert⁵⁰ und liegen in der 3. Normalform⁵¹ vor.

Die meisten Daten und Tabellen wurden allerdings von den Partnern vorgegeben, wodurch auf deren Daten nur ein geringer Einfluss bestand und dadurch auch für komplexere Strukturen keine ER-Modelle vorliegen.

Ein großes Problem stellten Daten-Updates der Partner dar, denn diese hatten oft nicht nur Aktualisierungen der Daten zur Folge, sondern auch Erweiterungen des Datenbestandes um zusätzliche Felder oder gar Tabellen. Das machte es nötig, flexibel auf die neuen Daten reagieren zu können, und gab den Anstoß eine eigene Skript-Sprache zu erstellen, die im zweiten Kapitel Techniken behandelt wird.

Der konkrete Aufbau der einzelnen Datenbanken wird im Kapitel sieben „Datenbanken, die Basis des Systems“ beschrieben.

⁴⁸ **Zugriffsrechte** sind Rechte, die den Zugriff auf Daten für bestimmte Nutzergruppen erlauben oder verhindert.

⁴⁹ In einem **ER-Diagramm** oder ER-Modell (entity / relationship model bzw. Gegenstands- / Beziehungsmodell) bildet man die Entitäten (abstrahierte Gegenstandstypen) und Relation (Beziehungen zwischen den Entitäten) ab. Damit wird versucht das gedankliche Konzept der modellierten Welt darstellbar zu machen.

⁵⁰ „Unter **Normalisierung** versteht man das Aufteilen der Daten in Relationen in der Art und Weise, dass sie am Ende den Normalisierungsregeln entsprechen. Ziel der Normalisierung ist in erster Linie eine Eliminierung von Redundanzen u. damit ein verständliches Datenmodell.“ [4, S.18 LE 4 (603)]

⁵¹ Die **3. Normalform** (3NF): „Eine Relation R ist in der dritten Normalform (3NF), wenn sie sich in der ersten und der zweiten Normalform befindet und kein Sekundärattribut (Nicht-Schlüsselattribut) transitiv abhängig von einem Schlüsselattribut ist.“ [4, S.23 LE 4 (603)]

1.3 Software-Ergonomie

Die Software-Ergonomie hilft mit ihren Vorgaben psychologische Belastungen zu verhindern. Das weltweit verfügbare Wissen zur ergonomischen Gestaltung von Software wird gesammelt, bewertet und in einer Norm integriert. Diese Norm bietet Hilfestellungen bei der Konzeption, Gestaltung und Bewertung von Bildschirmarbeitsplätzen und definiert Mindestforderungen für die ergonomische Gestaltung von Software.

Die Grundsätze der Software-Ergonomie sind in der für den deutschen Raum gültigen Norm DIN EN⁵² ISO⁵³ 9241 aufgeführt, die folgende Punkte umfasst:

Teil	Titel	Kategorie
9241-1	Allgemeine Einführung	
9241-2	Anforderungen an die Arbeitsaufgaben - Leitsätze	Allgemeiner Leitsatz
9241-3	Anforderungen an visuelle Anzeigen	Hardware
9241-4	Anforderungen an Tastaturen	Hardware
9241-5	Anforderungen an Arbeitsplatzgestaltung und Körperhaltung	Arbeitsumgebung
9241-6	Anforderungen an die Arbeitsumgebung	Arbeitsumgebung
9241-7	Anforderungen an visuelle Anzeigen bezüglich Reflexionen	Hardware
9241-8	Anforderungen an Farbdarstellungen	Hardware
9241-9	Anforderungen an Eingabegeräte außer Tastaturen	Hardware
9241-10	Grundsätze der Dialoggestaltung	Allgemeiner Leitsatz, Software
9241-11	Anforderungen an die Gebrauchstauglichkeit - Leitsätze	Allgemeiner Leitsatz
9241-12	Informationsdarstellung	Software
9241-13	Benutzerführung	Software
9241-14	Dialogführung mittels Menüs	Software
9241-15	Dialogführung mittels Kommandosprachen	Software
9241-16	Dialogführung mittels direkter Manipulation	Software
9241-17	Dialogführung mittels Bildschirmformularen	Software

Tabelle 1 – Grundsätze der Software-Ergonomie

⁵² Das **D**eutsche **I**nstitut für **N**ormung **DIN** ist seit 1975 per Vertrag mit der Bundesregierung die herausgehobene Normungsorganisation in Deutschland. Das DIN ist verpflichtet, die europäischen Normen zu übernehmen (**DIN EN**).

⁵³ „Die Internationale Organisation für Standardisierung **ISO** (**I**nternational **O**rganization for **S**tandardization) ist eine weltweite Vereinigung. Sie entstand im Jahr 1947 mit dem Ziel, die Standardisierung voranzutreiben und auf diese Weise sowohl den internationalen Austausch von Gütern u. Dienstleistungen als auch die Entwicklung der Zusammenarbeit im intellektuellen, wissenschaftlichen, technischen u. wirtschaftlichen Bereich zu erleichtern.“ [17]

1.3.1 Grundsätze der Dialoggestaltung

Teile zehn bis 17 der „Grundsätze der SW-Ergonomie“ enthalten konkrete Anforderungen an die ergonomische Gestaltung von Software. Teil zehn (Grundsätze der Dialoggestaltung) enthält die folgenden sieben Grundsätze, die unabhängig von einer bestimmten Software als allgemeine Leitlinien bei der Leistungsbeschreibung, Gestaltung und Bewertung von Dialogsystemen angewendet werden sollten:

1. Aufgabenangemessenheit⁵⁴
2. Selbstbeschreibungsfähigkeit⁵⁵
3. Steuerbarkeit⁵⁶
4. Erwartungskonformität⁵⁷
5. Fehlertoleranz⁵⁸
6. Individualisierbarkeit⁵⁹
7. Lernförderlichkeit⁶⁰

1.3.2 Anforderungen an die Gebrauchstauglichkeit

Teil elf der „Grundsätze der SW-Ergonomie“ definiert, was Benutzungsfreundlichkeit für Bildschirmarbeitsplätze heißt. Es wird darauf verwiesen, dass Benutzungsfreundlichkeit nur in einem größeren Zusammenhang gesehen werden darf. Nicht allein die Benutzungsfreundlichkeit der SW, sondern auch die Arbeitsumgebung, die Aufgaben u. Ziele des Benutzers, die verwendeten Arbeitsmittel und das soziale Umfeld sind für die Zufriedenheit des Benutzers ausschlaggebend. Um die gestellten Arbeitsziele zu erreichen, muss beurteilt werden, in welchem Maß diese Ziele erreicht wurden, welcher Aufwand dafür betrieben worden ist und wie zufrieden der Benutzer bei der Arbeit mit der Software war. Die Zielerreichung setzt sich aus den folgenden drei Kriterien zusammen: *Effektivität* (Qualität der Ergebnisse u. der Durchlaufzeiten), *Effizienz* (Fehlerhäufigkeit u. Korrekturzeiten) und *Zufriedenheit* (Benutzerbefragungen).

⁵⁴ Ein Dialog, der den Benutzer unterstützt, seine Arbeitsaufgabe effektiv und effizient zu erledigen, ist **aufgabenangemessen**.

⁵⁵ Alle Texte, wie Labels u. Meldungen, sollten auf Anhieb verständlich, also **selbstbeschreibend** sein.

⁵⁶ Schaltflächen, Icons und Menüeinträge sollten dem Benutzer mit einfachen und flexiblen Dialogwegen zum Ziel seiner Aufgabe führen und damit die Anwendung **steuerbar** machen.

⁵⁷ Bedienungsabläufe, Symbole u. die Anordnung von Informationen sollten innerhalb der Anwendung konsistent sein, dem erworbenen Wissen der Benutzer entsprechen, also **erwartungskonform** sein.

⁵⁸ In allen Situationen sollten Eingaben rückgängig gemacht und Bedienungsschritte aufgehoben werden können um damit das Programm **fehlertolerant** zu machen.

⁵⁹ Fenstereinstellungen, Spaltenanordnungen in Listen, Sortierungen, Symbolleisten, Menüs, Tastenkürzel etc. sollten individuell eingestellt u. gespeichert werden können, also **individualisierbar** sein.

⁶⁰ Alle Bedienungsschritte und „Orte“, wo bestimmte Informationen oder Funktionen zu finden sind, sollten einem leicht zu verstehenden und erlernbaren Prinzip folgen und daher **lernförderlich** sein.

Bis auf den Punkt sechs (Individualisierbarkeit) der „Grundsätze der Dialoggestaltung“ wurden in IMMOMENT die aufgeführten Punkte berücksichtigt und erfüllt. Durch die eingeschränkten Möglichkeiten der Benutzeroberfläche von Web-Seiten hätte der Aufwand den Nutzen überstiegen, dem Kunden Individualisierbarkeit zu ermöglichen. Der Nutzer kann nur durch den Browser eine individuelle Entscheidung über die Ansicht der Internet-Seiten von IMMOMENT treffen.

Sowohl die Fensterauschnittsgröße als auch die Größe der Schrift sind vorgegeben, damit nicht nur die Bilder, sondern auch die Texte möglichst in einem Bildschirmausschnitt (ohne scrollen) einheitlich angezeigt werden können. So kann der Nutzer im Rahmen des Unterpunktes „Anforderungen an die Gebrauchstauglichkeit“ seine Arbeitsumgebung nicht anpassen, denn diese ist optimiert auf einen Bildschirmausschnittsgröße von $1024 * 768^{61}$ Bildpunkten. Die Rücksprache mit den Kunden ergab, dass effizient mit dem Programm gearbeitet werden kann, denn bei IMMOMENT wurde berücksichtigt, dass der Nutzer bis zum Erhalt einer Information möglichst wenig Benutzungsschritte⁶² durchlaufen muss. Allerdings muss an der Effektivität der Informationen noch gearbeitet werden. Einige Partner können derzeit nur in bestimmten Bundesländern eine sehr hohe Informationsqualität liefern, das wird noch Thema in den Kapiteln sein, in denen die Dienste genauer vorgestellt werden.

⁶¹ **1024 * 768 Bildpunkte** stellte, wie Befragungen der meisten Interessenten und Kunden ergab, den zu dem Zeitpunkt üblichen Standard der Arbeitsplätze dar.

⁶² **Benutzungsschritte** sind Arbeitsschritte des Nutzers, bei denen die Rechneranwendung benutzt wird – also Eingaben über die Tastatur oder Maus (Auswahl eines Menüpunktes).

1.4 Trennung von Daten und GUI⁶³

Bei der Erstellung von Software, die auf verschiedenen graphischen Oberflächen eingesetzt werden kann und viele Daten verwaltet, wird diese Trennung auch in der Architektur der Software berücksichtigt. Aus diesem Grunde sollte eine Software wenigstens aus zwei Teilen oder Schichten bestehen, der Daten- und der GUI-Schicht.

Üblicherweise geht man heutzutage einen Schritt weiter und teilt die Software in wenigstens drei Schichten auf, die so genannte 3-Tier-Architecture⁶⁴. Diese Unterteilung hat den Vorteil, dass die Aufgaben auf verschiedene Server (Plattformen) verteilt werden können und die Wiederverwendbarkeit bzw. Austauschbarkeit einzelner Schichten (andere GUI oder anderer Datenbankserver) der Software möglich ist.

Anwendungen, die nach der 3-Tier-Architecture aufgebaut sind, eignen sich sehr gut dazu, eine „Thin Client“ Architektur abzubilden. Ein „Thin Client“ zeichnet sich durch einen schlanken Client (bei Web-Applikationen ist meist nur ein Browser nötig), einen oder mehrere Applikations-Server(n) und eine meist serverseitige Datenbank, die getrennt von den Applikations-Servern verwaltet wird, aus.

⁶³ Graphische Benutzeroberflächen (GUI - graphical user interface) unterstützen neben der Tastatur auch die Steuerung per Maus – z.B. Browseransicht oder Windowsoberfläche.

⁶⁴ **3-Tier-Architecture** (3-Schichten-Architektur). Dabei wird neben der Daten- (Datenverwaltung) und der Präsentationsschicht (GUI) noch eine dritte eingeführt, der Anwendungskern (Business Logic). Die oberste Schicht (Präsentation – Front-End oder Benutzerschnittstelle) stellt die Daten visuell dar und ermöglicht die Interaktion zwischen Anwendung und Nutzer. Die unterste Schicht oder Basis stellt die Datenverwaltung dar und als so genanntes Back-End des Systems übernimmt es das Speichern und Laden, also die Bereitstellung von Informationen.

Der Anwendungskern oder Middle-Tier (mittlere Schicht) realisiert die Fachlichkeit des Systems und muss streng von den anderen beiden Schichten getrennt werden, um nicht durch vermeidbare Abhängigkeiten die Konzepte der Wart- und Wiederverwendbarkeit zu verhindern.

1.5 Arten von Internetseiten

Internetseiten lassen sich in drei Gruppen aufspalten (entsprechend dem E-Government-Handbuch des BSI⁶⁵):

- statische Internetseiten (reine HTML-Seiten)
- Internetseiten mit „aktiven Inhalten“ - werden clientseitig ausgeführt
- dynamische Internetseiten - werden serverseitig ausgeführt

Das HTTP-Protokoll ist statusfrei, das bedeutet, dass jede Anfrage unabhängig und nicht über vorherige Anfragen informiert ist. Es ist jedoch oft notwendig, Statusinformationen zu speichern. Der Inhalt des Einkaufswagens eines Online-Shops sollte zum Beispiel jeder Anfrage bekannt sein. Werden nur *statische Internetseiten* eingesetzt, ist die Speicherung von z.B. Nutzer-Eingaben nicht möglich.

Es werden Technologien angeboten, so genannte „*aktive Inhalte*“, die es ermöglichen, bereits auf dem Client-Rechner speziell auf Nutzer-Eingaben zu reagieren bzw. komplett eigenständige Programme ablaufen zu lassen. Die wichtigsten Beispiele sind „Javascript“, „VBscript“, „Java⁶⁶“ und „ActiveX⁶⁷“ sowie Plugins⁶⁸.

⁶⁵ Aus dem E-Government-Handbuch des **BSI** (Bundesamt für Sicherheit in der Informationstechnik): „Unter ‚aktiven Inhalten‘ sind Computerprogramme zu verstehen, die in Internet-Seiten enthalten sind oder beim Betrachten einer Internet-Seite automatisiert nachgeladen werden. Ausgeführt werden diese Computerprogramme auf dem Computer des Internet-Nutzers. Es ist davon auszugehen, dass aktive Inhalte, die von Behörden bereitgestellt sind, ausreichend getestet und sicher sind. Aber dem Bürger kann ein Schaden entstehen, wenn er von einer Behördenseite mit aktiven Inhalten auf andere Internet-Seiten mit schädlichen Inhalten surft. Folglich sollte vermieden werden, dass der Bürger durch die Behördenseite gezwungen wird, aktive Inhalte zu akzeptieren. Von aktiven Inhalten grundsätzlich zu unterscheiden sind ‚dynamische Internet-Seiten‘. Um Suchfunktionen und andere flexible Angebote realisieren zu können, sind statische Internet-Seiten nicht ausreichend. Abhängig von Eingaben des Benutzers müssen Internet-Seiten dynamisch erzeugt, aufbereitet und an den Benutzer übertragen werden. Diese Funktionen lassen sich durch Programme und Skripten realisieren, die auf dem WWW-Server ausgeführt werden.“ [18]

⁶⁶ **Java** ist eine plattformübergreifende Objektorientierte-Programmiersprache, entwickelt von Sun. Wenn im Browser eine Seite mit einem Java-Programm angewählt wurde, wird ein so genanntes Applet übertragen und im Kontext des lokalen Browser ausgeführt.

⁶⁷ **ActiveX** ist eine von MS entwickelte Technologie, die aus mehreren Komponenten besteht. Sie können als Programme aufgerufen oder in einer WWW-Seite dargestellt werden u. enthalten Maschinencodes. ActiveX-Controls mit Maschinencode laufen direkt auf dem Rechner ab u. haben Zugriff auf Hardware und Betriebssystem. Bei **ActiveX** ist das Sicherheitsrisiko unübersehbar, da Zugriffe auf alle Funktionen des Rechners möglich sind. Verschiedene Zertifizierungsmethoden u. einstellbare Sicherheitsebenen ermöglichen eine Verminderung der Risiken.

⁶⁸ **Plugins** sind Programme, die sich in den Browser integrieren und diesen dann um Datei-Formate erweitern, die der Browser nicht selbst anzeigen kann. Bekannte Plugins: Pdf, Flash oder Realplayer. Plugins müssen zwar installiert werden, laden danach aber automatisch die Dateien, die von ihnen angezeigt werden können. Plugins können zu Problemen führen, sie haben vollen Zugriff auf Ressourcen des Rechners (mit Rechten, die auch jedes andere vom Nutzer gestartete Programm hat).

„Aktive Inhalte“ werden automatisch geladen, wenn das nicht durch Sicherheitseinstellungen des Internet-Browsers unterbunden ist. Ausgeführt werden sie entweder vom Internet-Browser oder von dem darunter liegenden Betriebssystem. Um Informationen clientseitig zu speichern, können auch Cookies⁶⁹ benutzt werden.

Es gibt noch eine Möglichkeit clientseitig Informationen von einer Anfrage zur nächsten zu sichern, durch so genannte Hidden-Fields⁷⁰.

ASP.Net bietet mit dem so genannten Viewstate⁷¹ eine erweiterte Form der Hidden-Fields. Wird eine Internet-Seite (WebForm) erstellt, so wird automatisch für jedes WebControl⁷² in einem Viewstate die dort eingetragene Information gespeichert und nach der nächsten Anfrage aus dem Viewstate das entsprechende WebControl wieder gefüllt.

Da dieser „Service“ die zu übertragende Seite vergrößert, sollte der Viewstate für Felder, deren Informationen später nicht benötigt werden, ausgeschaltet werden.

⁶⁹ „Die **Cookie**-Technik (Cookie, engl. für Kekse) erlaubt es einem Web-Server, auf dem PC des Anwenders Informationen zu hinterlegen. Da der Web-Server nicht direkt auf die Datenträger der Anwenders Zugriff hat, muss dies vom Browser erlaubt sein. Es können aber nur Textinformationen geschrieben werden, z.B. aktuelle Adresse, Anwenderrechner, Anwendereingaben. Beim nächsten Aufruf eines solchen Web-Servers werden die für ihn gültigen Daten aus der Cookie-Datei übertragen. Bei einigen Webservern können deren Angebote nur dann genutzt werden, wenn man die Annahme von Cookies zulässt.“ [19] Mit dieser Technik könnte aber auch ein Profil über Surfgewohnheiten des Anwenders erstellt werden. Stichwort: „Der gläserne Anwender“. Ein direktes Sicherheitsrisiko stellt die Cookie-Technik allerdings nicht dar.

⁷⁰ **Hidden-Field** (engl. bedeutet „verstecktes Feld“) ist ein Element der Beschreibungssprache HTML. Wie der Name schon sagt, sind sie auf der Internet-Seite nicht sichtbar, sie können Informationen speichern, die beim Aufruf der selben Internet-Seite (so genanntes „postback“) „weitergereicht“ und somit wieder angezeigt werden.

⁷¹ Das HTML-Tag `<input type="hidden" name="__VIEWSTATE" value="dDw3NTI5MDg7Oz4=" />` stellt einen so genannten **Viewstate** dar. Das obige Beispiel enthält den Eintrag eines Textfeldes. Der komplette Textinhalt des Feldes wird im value-Eintrag gespeichert. Der Eintrag des Viewstate scheint verschlüsselt, es wird aber nur das global bekannte Base64-Format verwendet, das sicherstellt soll, dass auch Sonderzeichen nach einem HTTP-Requests zum Server, korrekt wieder angezeigt werden können. Falls sicherheitsrelevante Daten im Viewstate gespeichert werden, können diese statt mit base64 auch mit TripleDES oder ähnlichen Verschlüsselungen geschützt werden. Grundsätzlich ist bei Erstellung von Internetseiten der Viewstate aktiv, das bedeutet, dass alle variablen Informationen der einzelnen WebControls des ASP.Net in den Viewstates gespeichert werden. Ist das nicht gewünscht, so muss für jedes WebControl, oder eine komplette Internet-Seite der Viewstate abgeschaltet werden. Nachtrag: **Base64**-Format – ursprünglich beim Übertragen von eMails benutztes Umsetzungsformat von Attachments, damit auch 8-bit Sonderzeichen über 7-bit Strecken übertragen werden konnten.

⁷² **WebControls** sind serverseitige Steuerelemente unter ASP.Net. Diese ermöglichen dem Client sowohl maus- und tastaturgesteuerte Eingaben (z.B. Textfelder oder Button), als auch Ausgaben von Datenbankwerten anzuzeigen, wofür häufig das Datagrid (Tabellenstruktur) verwendet wird.

Die Inhalte der einzelnen Internet-Seiten (bzw. Teile der Internet-Seiten) werden bei „*dynamisch generierten Internet-Seiten*“ von Programmen aufbereitet bzw. in einer Datenbank gespeichert und bei Aufruf einer Internet-Seite in diese eingefügt. Bekannte Vertreter sind: Servlets⁷³, CGI⁷⁴, ISAPI⁷⁵, PHP⁷⁶, ASP⁷⁷, bzw. ASP.Net.

Während aktive Inhalte Risiken für den Nutzer (Client) bedeuten, stellen dynamische Internet-Seiten Gefährdungen für den Betreiber (Server) dar.

Zusammenfassend kann festgehalten werden, dass bei Konzeption und Erstellung von Internetseiten der Einsatz von „Aktiven Inhalten“ möglichst gering gehalten werden sollte. Internetseiten mit „Aktiven Inhalten“ stellen auf dem Client-Rechner ein großes Sicherheitsrisiko durch ihre uneingeschränkten Rechte auf dem System dar. Insbesondere wenn zum Kundenkreis Banken gehören, werden ActiveX-Komponenten obligatorisch abgelehnt, aber auch Cookies sind durch herstellbare Anwenderprofile (Schlagwort: „gläserner Kunde“) nicht gern gesehen. Hingegen kann der Einsatz von Scripten (Javascript oder VB-Script) innerhalb einer Web-Anwendung kaum umgangen werden.

⁷³ **Servlets** sind das serverseitig Gegenstück zu den clientseitigen Java-Applets.

⁷⁴ **CGI** (Common Gateway Interface) beschreibt ein spezielles Protokoll zwischen Client-Browser und Webserver. Dadurch startet der Client-Browser auf dem Webserver ein Programm, welches eine Internetseite generiert und diese an den Client-Browser zurückschickt.

⁷⁵ **ISAPI** (Internet Server Application Programming Interface). Statt ausführbare Dateien mit Funktionalität auszustatten (wie bei CGI), verwendet ISAPI dll (dynamic link library) Dateien. ISAPI teilt den Speicher besser auf. Dadurch kann der Server mehr Anfragen als CGI gleichzeitig bearbeiten.

⁷⁶ **PHP** steht offiziell für "Hypertext Preprocessor". PHP ist eine Scriptsprache zur dynamischen Erstellung von Webseiten. Die Anweisungen der Sprache sind dabei in den HTML-Code einer Webseite eingebettet, d.h. jede HTML-Seite ist auch ein gültiges PHP-Programm. Die Syntax von PHP ist ähnlich wie die von C, Java oder JavaScript. Die Sprache zeichnet sich vor allen Dingen durch ihre leichte Erlernbarkeit, ihre ausgezeichneten Datenbankanbindungen u. Internet-Protokolleinbindungen und die Unterstützung zahlreicher weiterer Funktionsbibliotheken aus. PHP stellt so für den Web-Entwickler das ideale Werkzeug zur Erstellung von dynamischen Inhalten dar. PHP ist freie Software im Sinne der Debian Free Software Guidelines (DFSG). Quelltext und Binaries des PHP-Interpreters sind frei erhältlich und können für alle kommerziellen u. nichtkommerziellen Zwecke eingesetzt werden; jeder kann den PHP-Quelltext weiterentwickeln u. die Änderungen an das PHP-Projekt zurückfließen lassen. Der genaue Lizenztext ist in der Datei license-php.txt enthalten, die Bestandteil der PHP-Distribution ist. PHP läuft auf allen gängigen Unix-Versionen und auf den verschiedenen Windows-Versionen (Windows 95/98/ME/NT/2000). Als CGI-Programm kann PHP mit jedem Webserver zusammenarbeiten. Für einige Webserver, allen voran Apache (lizenzfreier Open Source Webserver), stehen auch Modulversionen zur Verfügung, die sehr viel effizienter ausgeführt werden.“ [20]

⁷⁷ **ASP** (Active Server Pages) ist keine Programmiersprache, sondern eine von MS entwickelte Umgebung für interaktive Internetseiten. ASP läuft dabei vollständig serverseitig ab und sendet reinen HTML-Code an den Browser. Somit ist ASP browserunabhängig. In ASP werden Skriptsprachen verwendet. Die zumeist verwendeten Skriptsprachen sind JavaScript u. VBScript, standardmäßig wird VBScript verwendet. Die Möglichkeiten von ASP sind sehr groß. Das Verwenden von Datenbanken ist eine der zumeist genutzten Möglichkeiten von ASP. Beispiel Gästebuch: Erstellt jemand einen neuen Eintrag in das Gästebuch, muss dieser Eintrag beim nächsten Aufruf des Gästebuches zur Verfügung stehen. Dabei können die unterschiedlichsten Datenbanken in ASP eingesetzt werden, von einer einfachen XML-Datei, einer Access-Datenbank, bis zu einer mehreren Gigabyte großen Datenbank, z.B. SQL-Server oder Oracle. Newsletter, Emailversand, Dateien-Upload, verarbeiten von Office-Dokument, Passwort-gesteuerter Zugriff sind nur einige wenige Möglichkeiten von ASP. ASP ermöglicht die Nutzung von ActiveX (exe u. dll), geschrieben in Visual Basic u. C++.“ [21]

In IMMOMENT wurden zur Speicherung und Anzeige von Informationen fast ausschließlich „dynamische Web-Seiten“ eingesetzt. Dieses Vorgehen wird aus Sicherheitsgründen vom Nutzer begrüßt, hat aber den Nachteil, dass zur Weiterbearbeitung grundsätzlich vom Client eine Anfrage an den Server gestellt werden muss, worunter die Verarbeitungsgeschwindigkeit leiden kann. Weitere Punkte in diesem Zusammenhang, wie die Verwaltung einer so genannten Session, werden im siebten Kapitel „Datenbanken, die Basis des Systems“ behandelt.

Um die Verarbeitungsgeschwindigkeit einiger Abläufe zu beschleunigen, wurden auch Javascripte eingesetzt, die es ermöglichen, direkt auf eine Anfrage des Clients zu reagieren. Hauptsächlich werden durch Javascripte „Doppelclicks“ abgefangen. Hat ein Nutzer einen Dienst gestartet, kann es vorkommen, dass dieser einige Sekunden benötigt, um die Daten auf einer weiteren Internet-Seite anzeigen zu können. Die Bearbeitung wird dem Nutzer durch ein farblich hervorgehobenes Textfeld auf der Internet-Seite angezeigt, auf der der Button ausgewählt wurde, der Text lautet: „Bitte warten Sie wenige Sekunden ...“. Übersieht der Nutzer den Text und klickt den Button ein weiteres Mal, erscheint eine Textbox, die bestätigt werden muss und der zu entnehmen ist, dass der Vorgang bereits bearbeitet wird. Javascripte werden zusätzlich auf den Seiten, die Bilder anzeigen, benötigt, um festzustellen, ob die Bilder vollständig im Browser angezeigt wurden. Ist das der Fall, so wird durch das clientseitige Javascript ein serverseitiges Programm ausgelöst, das in einer Protokoll-Tabelle des Datenbank-Servers einen Eintrag vornimmt.

Es werden auch Viewstates eingesetzt, die haben den Vorteil, dass ASP.Net die Verwaltung zu sichernder Texte selbst übernimmt und kein Eingriff vom Entwickler nötig ist, damit Daten nach einem Zugriff auf den Server wieder anzuzeigen.

Neben Javascripts werden vom Nutzer auf einigen Erklärungsseiten nur die Plugins des pdf-Readers bzw. des Entpackprogrammes Winzip erwartet, sonst werden keine weiteren Internetseiten mit „aktiven Inhalten“ eingesetzt. Der noch im Pflichtenheft beschriebene Einsatz einer ActiveX-Komponente, die die Anzeige der Straßenbilder (City-View und City-Server) ermöglicht (s. „Anhang A - 3.1 Software“), wird nicht mehr grundsätzlich benötigt, da der Partner die Bilder inzwischen wahlweise auch im JPEG⁷⁸-Format liefert, die ein Internetbrowser direkt anzeigen kann.

⁷⁸ **JPEG** (Joint Photographics Experts Group) Name des Komitees, das um Standardisierung bei Bild-Kompression bemüht war. Ermöglicht im verlustbehafteten Verfahren Reduzierung von Datenmengen (Bildern) stufenweise bis um den Faktor 25. JPEG ist als ISO-Norm akzeptiert.

1.6 Sicherheit von Internetseiten

Um Daten gesichert übertragen zu können werden im Internet für die gesicherte Kommunikation grundsätzlich zwei Verschlüsselungsverfahren eingesetzt: Die symmetrische und die asymmetrische Verschlüsselung.

Bei der symmetrischen Verschlüsselung müssen beide Kommunikationspartner im Besitz des gleichen Schlüssels sein. Gerät der Schlüssel dabei in die Hände eines Dritten, ist es für diesen relativ einfach, die übertragenen Daten zu decodieren. Aus diesem Grund hat sich der Standard S/MIME⁷⁹ (digitale Signatur⁸⁰) bei E-Mail und SSL⁸¹ bei Online-Transaktionen durchgesetzt.

Beide Verfahren unterstützen asymmetrische und symmetrische Verschlüsselung auf Basis von X.509-Zertifikaten⁸². Hauptziele von X.509-Zertifikaten sind zum einen die Authentifikation der miteinander kommunizierenden Stationen und zum anderen die Gewährleistung der Vertraulichkeit der Daten.

⁷⁹ „**MIME** (Multipurpose Internet Mail Extensions) ist ein internationaler E-Mail-Standard, der die Versendung von Dateien beliebiger Anwendungen erlaubt (Binärdateien), zum Beispiel für 8-bit-Grafiken, Video und Sounddateien usw. Das Internet kann 'nur' 7-bit-ASCII-Code verarbeiten. Solche '8-bit-Dateien' bzw. Binärdateien werden beim Verfassen eines Emails mittels Anhang oder Attachment an das Email 'angehängt' und dann von MIME in 7-bit-ASCII-Code codiert. Beim Empfänger decodiert MIME den Anhang wieder von 7-bit zu 8-bit-Code. MIME ist die Weiterentwicklung von UUencode/UUdecode - ursprünglich entwickelt für UNIX-Systeme. S/MIME ist nun eine Erweiterung zu diesem MIME-Standard, der zur sicheren Kommunikation über E-Mail ein PKI-Verfahren auf Basis von X.509-Zertifikaten nutzt. Nach **S/MIME** können Daten elektronisch unterschrieben oder verschlüsselt werden.“ [22]

⁸⁰ „**Digitale Signatur** bedeutet soviel wie „elektronische Unterschrift“ (Signieren = elektronisch unterschreiben). Wie im normalen Schriftverkehr werden Signaturen benutzt, um den Autor eines E-Mails oder von elektronischen Transaktionen eindeutig zu bestimmen. Zu dieser Funktionalität, die eine Unterschrift ja auch im normalen Schriftverkehr leistet, können mit einer elektronischen Unterschrift auch Veränderungen an E-Mails oder Transaktionen festgestellt werden.“ [22]

⁸¹ „**SSL** (Secure Sockets Layer) ist ein offener Standard der Firma Netscape für die gesicherte Datenübertragung im Internet. Mit Hilfe von SSL wird der unberechtigte Zugriff auf sicherheitsrelevante Informationen wie etwa Kreditkartennummern verhindert um einen sicheren elektronischen Zahlungsverkehr über das World Wide Web zu ermöglichen. Für die Verschlüsselung der Daten wird keine spezielle Software benötigt, sondern lediglich ein normaler Browser mit Sicherheitsfunktion. Die Verschlüsselung von Informationen ist am genutzten Protokoll "https", das der Internetadresse vorangestellt ist, zu erkennen (z.B. <https://www.trustedshops.de>). Des Weiteren erscheint ein Schloss- oder Schlüsselsymbol im Browserfenster.“ Vergleiche [23]

⁸² „**X.509-Zertifikat** - Zertifikate sind mit Ausweisen wie Führerschein oder Reisepass vergleichbar. Der Internet-Standard X.509 beschreibt die Form eines Software-Zertifikats, es ist ein Standardformat der ITU-T für Zertifikate. Es beinhaltet den Namen und die digitale Signatur des Ausstellers sowie Informationen über die Identität des Inhabers. Mit X.509-Zertifikaten kann man auf den gesicherten Bereich einer Web-Seite zugreifen oder E-Mails unterzeichnen u. verschlüsseln.“ [22]
ITU-T (International Telecommunications Union - Telecommunication Sector): Teil-Organisation der ITU, die sich mit der Standardisierung im internationalen Fernsprechbereich befasst. Die ITU ist eine Unterorganisation der UNO (United Nations Organisationen).

Unter Authentifikation der Teilnehmer versteht man das Sicherstellen der Identität der Parteien, also das Überprüfen, ob die Daten auch wirklich von der Person stammen, die diese versendet haben soll. Dies wird durch ein so genanntes Zertifikat eines "Certification Authority" (CA), einer anerkannten Beglaubigungsstelle, gewährleistet.

Es wird mit der asymmetrischen Verschlüsselung erst ein gemeinsamer Anwendungsschlüssel (session key) ausgetauscht, mit dem dann die Kommunikation symmetrisch verschlüsselt wird.

Derzeit stellt das Zwei-Schlüssel-System PKI⁸³ die gängigste Lösung dar, um im offenen Internet sensible, schutzbedürftige Daten vertraulich zu übertragen.

Die üblichen Internet-Browser-Programme unterstützen SSL und S/MIME. Durch die Verwendung von SSL wird ein Kommunikations-Kanal zwischen dem Web-Browser und dem Server aufgebaut, der das Eindringen Dritter abblockt. Ist diese Verbindung einmal aufgebaut, so können schutzbedürftige Informationen wie Kreditkarten-Nummern ohne Zugriff von Unbefugten übertragen werden.

Nachteil von SSL ist, dass die Übertragungsgeschwindigkeit der Daten, insbesondere bei vielen zu übertragenden Daten (z.B. Bilder), deutlich herabgesetzt wird.

Die Web-Applikation IMMOMENT wird zukünftig per SSL ihre Daten gesichert übertragen, die Web-Services übermitteln bereits derzeit mit dem neuen Standard WS-Security (seit 12/2002), das eine Protokollergänzung darstellt, sehr sicher die Daten. Dieser Standard ermöglicht verteilten Applikationen, die das SOAP-Protokoll verwenden, eine sichere standardisierte Kommunikation über Web-Services zu Client-Applikationen. Das Vorgehen hat aber den Vorteil, dass ressourcensparend nur Teile des gesamten SOAP-Protokolls, die sensitive Informationen enthalten, verschlüsselt werden. Dies wird durch digitale Signatur (mit X.509-Zertifikaten) sowie Verschlüsselungsalgorithmen wie AES (Rijndael)⁸⁴ oder TripleDES⁸⁵ realisiert.

⁸³ **PKI** (Public Key Infrastructure) oder auch Zwei-Schlüssel-System genannt erzeugt für jeden Nutzer zwei digitale Schlüssel, die einmalig existieren und sich ergänzen. Der eine Schlüssel – „private key“ – ist der persönliche (private) Schlüssel und darf nicht bekannt werden, der andere – „public key“ – (öffentlicher Schlüssel) soll bekannt werden und wird (wie ein Telefonverzeichnis) vom Trust Center verwaltet. Will man jemandem eine vertrauliche Nachricht schicken, so holt man sich vom Trust Center den public key des Empfängers und damit wird die Information verschlüsselt. Der private key des Empfängers ist der einzige Schlüssel, der zu seinem public key die versendete Nachricht entschlüsseln kann, somit kann nur der Empfänger die Nachricht öffnen und kein anderer. Dabei dokumentiert die digitale Signatur zweifelsfrei den Versender und gewährleistet, dass die Daten auf dem Weg zum Empfänger nicht verändert wurden. Dazu wird der Hash-Wert (oder elektronischer Fingerabdruck - eine Art Prüfsumme) der Nachricht verschlüsselt. Passt dieser Hash-Wert nicht mehr zum Dokument, wurde dieses verändert.

⁸⁴ **AES** (Advanced Encryption Standard) mit der Chiffre Rijndael unterstützt die Schlüssellängen 128, 192 und 256 Bit. Dabei wird die Verschlüsselung wenigstens zehn mal (bei 128 Bit) durchlaufen.

⁸⁵ Bei **TripleDES** wird das DES-Verfahren (Data Encryption Standard) dreimal durchlaufen. DES verschlüsselt dabei einzelne Bits nach festen Regeln, es ist durch die geringe Schlüsselgröße von 56 Bit nicht mehr zeitgemäß. TripleDES erhöht die Sicherheit deutlich.

2 Techniken

In IMMOMENT werden verschiedene Techniken eingesetzt, die bestimmte Logik aus den Quelltexten in die Datenbank verlagern, damit bei Änderungen z.B. von Formulierungen der Textausgabe der Partner-Daten bzw. nach Daten-Updates keine Quelltext-Anpassung erforderlich ist, sondern diese Änderungen im Administrationsbereich über einen Internet-Browser vorgenommen werden können. Wie die Umsetzung durchgeführt wird, soll im ersten Unterkapitel „Die Skript-Sprache“ genauer dargestellt werden.

Ein weiteres Unterkapitel beschreibt einen Aspekt der Entwicklungsumgebung, sie unterstützt den Programmierer in vielen Bereichen, z.B. bei der „Aufteilung von Logik- und Präsentationsschicht“.

In IMMOMENT werden Bilder und in bestimmten Umkreisen um eine vorgegebene Adresse Informationen angezeigt. Dazu sind Lage-Koordinaten nötig. Welches Koordinaten-System in IMMOMENT Verwendung findet, wie und warum es genutzt wird, erläutern die beiden aufeinander folgenden Kapitel „Gauß-Krüger-Koordinaten“ und „Datensuche durch Lage-Koordinaten“.

Das letzte Unterkapitel beschäftigt sich mit dem Thema „Datenschutz“, denn in IMMOMENT werden auch Daten angezeigt, die aus datenschutzrechtlicher Sicht kritisch sind. Das Unterkapitel erläutert, wie das Persönlichkeitsrecht des Einzelnen gewahrt bleibt.

2.1 Die Skript-Sprache⁸⁶

IMMOMENT enthält mehrere Textblöcke, die durch bestimmte Dienstauftrufe (s. Kapitel acht) ausgegeben werden. Diese Textblöcke ändern sich mehrmals im Jahr, wenn von den entsprechenden Partnern, Daten-Updates geliefert werden. Durch die Updates kommen meist auch neue Attribute oder gar Tabellen hinzu, deren Werte an bestimmten Stellen in einem Text-Feld der Internet-Seite auszugeben sind.

Würden die Textausgaben direkt durch die C#-Quelltexte gesteuert, so wären für jedes Daten-Update Änderungen in den Quelltexten nötig, gefolgt vom Einspielen des Programm-Updates beim Provider. Das hätte nicht nur Programmieraufwand von Entwicklern, sondern auch Kosten beim Provider zur Folge.

Es bot sich an, die Ausgabe über Tabellen in der Datenbank zu steuern, denn dann wären die nötigen Anpassungen über Administrationsseiten nicht nur durch Entwickler möglich, wodurch der Provider gar nicht beansprucht würde. Es besteht aber das Problem, dass neben festen auch variable Texte ausgegeben werden müssen, die von der eingegebenen Adresse abhängen, bzw. dass auf die Ausgabe von Teilen dieser Textblöcke verzichtet wird, wenn bestimmte Werte nicht vorhanden sind.

Eine Skript-Sprache⁸⁷ kann diese Anforderungen erfüllen. Da nur wenige Befehle benötigt wurden, bot sich die Entwicklung einer eigenen Skript-Sprache an.

Die Skript-Sprache erkennt anhand einer Sonderzeichenfolge, ob an die Stelle dieser Sonderzeichen ein bestimmter Attributwert der Datenbank nachgelesen werden muss, oder ob ein Befehl ausgeführt werden soll. Das Skript selbst steht in einer so genannten Steuerungstabelle und wird Datensatz für Datensatz eingelesen und nach Interpretation der Skript-Sprache wird der Text in einer Internet-Seite ausgegeben. Folgender Beispiel-Skriptblock soll das Vorgehen veranschaulichen:

„von Hausnummer: ##/STRAB_Basis/STRAB_NummerVon#§“.

Die Sonderzeichen „##“ und „#§“ werden von der Skript-Sprache erkannt und durch Datenbankwerte ersetzt. Der interpretierte Text sieht dann wie folgt aus:

„von Hausnummer: 21“.

⁸⁶ „Bei einer **Skript-Sprache** (engl.: scripting language) wird eine Folge von Anweisungen, Skript genannt, zur Ausführung [z.B.] in einer Datei abgelegt. Skriptsprachen erlauben die Automatisierung von wiederkehrenden, zumeist systembezogenen Verwaltungsarbeiten, zum Beispiel das Einrichten von Benutzerkennungen, die Sicherung des Personalcomputers nach vorgegebenen Merkmalen usw. Im Zusammenhang mit dem Internet werden Skriptsprachen auch für die Erstellung von WWW-Anwendungen eingesetzt. Die zwei bekanntesten und populärsten Vertreter von Skriptsprachen stellen Perl und REXX dar, die beide für zahlreiche Betriebssysteme portiert wurden.“ [8]

Im Weiteren wird erläutert, wie das zu interpretierende Skript aufgebaut sein muss, damit der gewünschte Text ausgegeben wird und welche Fähigkeiten die in C# erstellte Skript-Sprache besitzt.

2.1.1 Aufgaben der Steuerungstabellen

Steuerungstabellen, die in der Datenbank verwaltet werden, regeln den Aufbau des auszugebenden Textes. Steuerungstabellen bestehen aus zwei Attributen⁸⁸:

2.1.1.1 Das Attribut „ID“

Die „ID“ enthält eine Zahl als Primärkey⁸⁹, das ermöglicht die Sortierung der Ausgabe. Da die Ausgabe in aufsteigender Reihenfolge der „ID“ erscheint, kann diese schnell geändert bzw. durch Sprünge in der Nummerierung können freie „ID“ Werte neu belegt und somit zwischen vorhandenen Blöcken Text eingefügt werden.

2.1.1.2 Das Attribut „zeile“

Das Attribut „zeile“ enthält das zu interpretierende Skript. Damit neben dem festen Text auch auf die aktuellen Daten für die entsprechende Adresseingabe reagiert werden kann, werden Steuerzeichen in dem Text verankert.

2.1.2 Theoretische Grundlagen

Eine Grammatik⁹⁰ prüft die syntaktische Korrektheit einer Zeichenfolge des Skriptes. Der syntaktischen Prüfung folgt ein spezieller „Automat“⁹¹ die „Turingmaschine“⁹²,

⁸⁷ Eine **Sprache** ist charakterisiert durch ein Regelwerk für diejenigen Zeichenketten, die in der Sprache formulierbar sind.

⁸⁸ Ein **Attribut** ist ein Feld bzw. eine Spalte in einer relationalen Datenbanktabelle.

⁸⁹ Ein **Primärkey** oder Primary Key (eindeutiger Index oder Schlüsselfeld) ist ein Attribut einer Tabelle, das einen Datensatz (Tupel) in dieser Tabelle dauerhaft eindeutig kennzeichnet. Index Einträge sind extra abgelegte Informationen von Spaltenwerten, über die die Datenbank schneller auf die Werte zugreifen kann (Vorsortierung). Man unterscheidet Primär und Sekundär Indexe. Es kann pro Tabelle nur einen Primär Index gegeben und beliebige Sekundärindexe.

⁹⁰ „**Grammatik** [...] nennt man die formalen Beschreibungsverfahren, mit denen man den strukturellen Aufbau einer formalen Sprache festlegt [...]. Die BNF-Notation (Backus-Nauer-Form), mit deren Hilfe in der Informatik oftmals weite Teile der Syntax einer Programmiersprache definiert werden, ist eine solche Grammatik. Formale Grammatiken wurden bereits 1956 von dem Linguisten Noam Chomsky eingeführt.“ [6, S. 5]

⁹¹ „**Automaten** im allgemeinen sind algorithmisch arbeitende Maschine, die zu gegebenen Eingaben ein bestimmtes „Programm“ abarbeitet und gewisse Ausgaben produziert.“ [6, S. 6] „Man unterscheidet akzeptierende (Syntaxprüfung) und berechnende Automaten (Berechnungsmaschinen), letztere werden auch „endliche Automaten“ genannt.“ [Vergleiche 6, S. 6]

⁹² Kellerautomaten sind „endliche Automaten“, die um einen „Kellerspeicher“ (Stapel oder Stack) erweitert wurden, damit sie sich z.B. Klammerebenen „merken“ können. Sind Manipulationen im Speicher an beliebigen Stellen nötig, so wird eine **Turingmaschine** (Gedankenmodell einer univiersellen Rechenmaschine von M. Turing) benötigt, die einen erweiterten Kellerautomaten darstellt.

die aufgrund bestimmter Zeichenfolgen erkennt, dass Textblöcke zu ersetzen sind.

Die Steuerungstabelle wird Datensatz für Datensatz eingelesen und nach jedem Einlesen eines Datensatzes wird dieser überarbeitet und danach ausgegeben.

Jedes Attribut „zeile“ wird von hinten nach vorn untersucht. Zuerst werden alle nachzulesenden Datenblöcke eingefügt und erst danach werden zu interpretierende Befehle, bzw. mathematische Operationen ausgeführt. Können keine Steuerzeichen entdeckt werden, so wird der Inhalt des Attributes „zeile“ direkt ausgegeben.

Sollte nach einer Startzeichenfolge (## oder #?) eine entsprechende Endezeichenfolge (§) fehlen, so wird eine Fehlermeldung dem Text angehängt (Syntaxcheck).

2.1.3 Erkennen von Werten („##“)

Damit die nachfolgend beschriebene Logik funktioniert, müssen zuerst für eine eingegebene Adresse bestimmte Daten nachgelesen werden. Diese Daten werden in einem XML-Strom hinterlegt.

Wird im Skript die Sonderzeichenkombination „##“ entdeckt, so wird das direkte Nachlesen von Werten aus dem vorbereitete XML-Strom eingeleitet.

Dabei wird eine Pfad-Darstellung erwartet, die in einfacher Form dem XPATH⁹³ von XML entspricht, z.B. bedeutet ##/haus/HAUS_Haushalte#§, dass aus der Tabelle „haus“ der Wert des Attributes „HAUS_Haushalte“ nachgelesen und direkt an dieser Stelle ausgegeben wird.

Es können auch verschachtelte Werte nachgelesen werden, dabei wird der Aufbau eines XML-Statements genutzt. Z.B. wird die folgende Angabe in zwei Schritten zerlegt: ##/h2_tabelle[ID=##/haus/HAUS_Haustyp#§]/text#§ da die Sonderzeichen von hinten nach vorn gesucht werden, wird zuerst der innerste (hintere) Wert nachgelesen, also ##/haus/HAUS_Haustyp#§. Das Ergebnis wird entsprechend eingesetzt (##/h2_tabelle[ID=X]/text#§) und nun kann der äußere Wert nachgelesen werden.

2.1.4 Erkennen von Befehlen („#?“)

Es ist auch möglich, nachgelesene Werte in mathematischen Operationen weiterzuberechnen oder durch IF – ELSE – ENDIF Konstrukte zu überprüfen, ob weitere Textstellen ausgegeben werden sollen. Befehle werden durch die Sonderzeichen „#?“

⁹³ **XPATH** (XML Path Language) ermöglicht den Zugriff auf einzelne Unterknoten in einem XML-Strom. Dabei wird die Baumstruktur des XML-Datenstromes, wie die Pfade in einem Dateisystem angesehen.

eingeleitet und ebenfalls mit „#§“ abgeschlossen. Beide Berechnungen werden direkt als SQL-Statement gegen die SQL-Server Datenbank abgefragt.

Für den folgenden Ausdruck soll eine mathematische Berechnung (Addition zweier Werte $X + Y$) durchgeführt werden:

```
#?MATH          ##/STRAB_Haustyp/STRAB_Haustyp_1#§      +
##/STRAB_Haustyp/STRAB_Haustyp_2#§ #§
```

Im ersten Schritt werden die Werte ##/STRAB_Haustyp/STRAB_Haustyp_1#§ und ##/STRAB_Haustyp/STRAB_Haustyp_2#§ wie bereits beschrieben nachgelesen und danach kann die Abfrage #?MATH $X + Y$ #§ interpretiert werden. Nun wird das SQL-Statement „SELECT $X + Y$ “ an den SQL-Server geschickt, der das Ergebnis zurückliefert, und der Textblock #?MATH $X + Y$ #§ wird durch das Ergebnis der Berechnung ersetzt.

Der Aufbau des IF-Befehls ist etwas komplexer, da als Rückgabe nur benötigt wird, ob das Ergebnis positiv (true) oder negativ (false) ist. Soll z.B. das IF-Konstrukt #?IF##/STRAB_Haustyp/STRAB_Haustyp_7#§ > 0#§ interpretiert werden, so wird im ersten Schritt wieder der Datenbankwert ##/STRAB_Haustyp/STRAB_Haustyp_7#§ nachgelesen und danach die IF-Abfrage #?IF $X > 0$ #§ als SQL-Befehl folgendermaßen aufgebaut:

```
SELECT CASE WHEN (X > 0) THEN 'true' ELSE 'false' END.
```

Das positive Ergebnis (true) bewirkt, dass die nachfolgenden Zeilen bis zum folgenden #?ENDIF#§ ausgegeben werden. Die Rückgabe „false“ bewirkt das Gegenteil. Ist ein #?ELSE#§ Zweig vorhanden, so wird die Ausgabe-Logik entsprechend bis zum folgenden #?ENDIF#§ umgekehrt.

Der IF-Befehl (ebenso ELSE und ENDIF) interpretiert nur, ob die nächste(n) Zeile(n) ausgegeben werden. Es können auch komplexe IF-Befehle interpretiert werden, dabei dürfen nur die Operatoren und Funktionen von SQL⁹⁴ benutzt werden, die SQL-Server akzeptiert, denn der Befehl wird zur Interpretation an die Datenbank abgeschickt.

Es zeigte sich, dass es auch notwendig ist, innerhalb einer auszugebenden Zeile zu interpretieren, ob ein bestimmter Teil ausgegeben werden soll oder ein anderer. Dieses Problem stellte sich insbesondere bei Plural- oder Singularformen in einem Satz. An folgendem Beispiel kann es besser erläutert werden:

```
Privatadress#?EXCELIF ##/haus/HAUS_Haushalte#§>1;en;e#§.
```

⁹⁴ **Operatoren und Funktionen von SQL** sind z.B.:

Operatoren: +, -, /, ||, AND, OR, NOT, =, >, IN, LIKE Funktionen: UPPER(), LENGTH(), ROUND()

Ist der Wert des Attributes „HAUS_Haushalte“ in der Tabelle „haus“ größer als 1, so soll die Endung „en“ an das Wort „Privatadress“ angehängt werden, ansonsten „e“. Um das Problem lösen zu können, wurde der wenn-Befehl⁹⁵ bzw. IF-Befehl aus MS-Excel⁹⁶ als EXCELIF umgesetzt.

Das Skript von IMMOMENT benötigt keine Anführungszeichen, um einen Textwert auszugeben, da alles, was ausgegeben wird, als alphanumerischer Datentyp interpretiert wird, eine Verschachtelungstiefe ist unbeschränkt möglich.

Mathematische Operationen mussten über die SQL-Server Datenbank abgefragt werden, da der Sprachumfang von C# keine Möglichkeit umfasst, eine beliebige Berechnung aus einem Text heraus durchführen zu können – andere Sprachen ermöglichen das direkt, z.B. stellt Javascript die Funktion eval() zur Verfügung.

Es wurde ein Formatierungsbefehl für die Textausgabe benötigt, so dass der Befehl TAB als Tabulator-Befehl eingebaut wurde, mit dem man eine feste Anzahl von Leerzeichen Abstand vom Rand erwirken kann. Folgende Zeile bedeutet, dass der Text „überwiegend wohnwirtsch. genutzte Objekte mit 1-2 HH“, mit einem Abstand von drei (3) festen Tabulatorsprüngen (sieben Leerzeichen) vom linken Rand (L) abgesetzt wird: #?TAB3Überwiegend wohnwirtsch. genutzte Objekte mit 1-2 HH:#§

Ein Möglichkeit Hilfetexte einzufügen wurde als notwendig erachtet, so wurde ein REM-Befehl umgesetzt (#?REM Text #§). Eine solche Zeile wird nicht ausgegeben.

2.2 Aufteilung von Logik- und Präsentationsschicht

Visual-Studio .Net ist eine Entwicklungsumgebung, mit deren Hilfe sowohl die Programm-Oberflächen, als auch die Quelltexte und das gesamte Programm erstellt (compiliert), oder per Debugger nach logischen Fehlern untersucht werden kann.

Die Entwicklungsumgebung bietet zusätzliche Möglichkeiten, z.B. unterstützt sie den Entwickler auch bei der strukturierten Erstellung einer Web-Applikation, indem die „Aufteilung von Logik- und Präsentationsschicht“ bereits durch Trennung in verschiedene Quelltext-Dateien vorbereitet wird.

⁹⁵ Die deutsche Hilfe in Excel beschreibt es wie folgt: „WENN(Prüfung; Dann_Wert; Sonst_Wert) – z.B. WENN(A10<=100; "Im Budget"; "Über Budget"). In einer Budgettabelle enthält Zelle A10 eine Formel zum Berechnen des aktuellen Budgets. Wenn das Ergebnis der Formel in Zelle A10 kleiner oder gleich 100 ist, zeigt die folgende Funktion "Im Budget" an. Andernfalls wird „Über Budget“ angezeigt. Damit es möglich ist, kompliziertere Bedingungen zu formulieren, können bis zu sieben WENN-Funktionen als Dann_Wert- und Sonst_Wert-Argumente geschachtelt werden.“

⁹⁶ MS-Excel ist das Tabellenkalkulationsprogramm von Microsoft

Wie bereits im ersten Kapitel unter „Trennung von Daten und GUI“ aufgezeigt wurde, teilt die heutige Software-Architektur Programme in Schichten auf.

Die Entwicklungsumgebung von ASP.Net unterstützt den Entwickler bei der Aufteilung zweier Schichten, der Präsentationsschicht und der Business Logic. Wird z.B. eine Web-Applikationen erstellt mit der Startseite „WebForm“, so werden für diese folgende Dateien angelegt: „WebForm.aspx“ und „WebForm.aspx.cs“ (je nach Anwendungsfall wird noch die Datei „WebForm.aspx.resx“ angelegt, die bestimmte Ressourcen enthält). Die Datei mit der Endung „.cs“ enthält dabei den Quelltext (oder Business Logic auch „code behind“ genannt), der in verschiedenen .Net-Programmiersprachen erstellt worden sein kann, in diesem Beispiel wurde die Programmiersprache C# gewählt, somit hat die Quelltextdatei die Endung „.cs“. Die Datei mit der Endung „.aspx“ enthält den gesamten HTML-Code (die Präsentationsschicht).

Bei der Erstellung der Oberfläche (GUI) unter ASP.Net wird der Programmierer durch eine große Anzahl serverseitige Steuerelemente (so genannte „WebControls“) unterstützt. ASP.Net arbeitet vollständig ereignisorientiert, d.h. Ereignisse, die der Benutzer im Browser auslöst, werden automatisch bis zum Server weitergereicht und dort an eine vom Programmierer definierte Ereignisbehandlungsroutine übergeben (meist im „code behind“).

Die aspx-Dateien werden nicht direkt interpretiert, sondern beim ersten Aufruf durch den Client in IL-Code⁹⁷ übersetzt. Auf dem Server hat nun die CLR⁹⁸ nicht nur die Aufgabe den IL-Code in Maschinensprache zu übersetzen (durch einen JIT⁹⁹-Compiler), sondern sie regelt auch die Speicherverwaltung (Garbage-Collection, Objektverwaltung, ...), verwaltet Prozesse und Threads, setzt Sicherheitsrichtlinien durch und lädt Komponenten (Assemblies). Ruft nun ein Client eine WebForm auf, so werden die Web Controls in normalen HTML-Code übersetzt und jede Benutzeraktion im Browser wird durch ein HTTP-Post an den Server übermittelt.

⁹⁷ Um von der verwendeten Programmiersprache unabhängig zu sein hat Microsoft **IL** (Intermediate Language) eingeführt, alle eingesetzten Programmiersprachen erzeugen diesen **IL-Code**. Dieser IL-Code stellt eine Zwischensprache dar, die alle Konstrukte (Klassen, Typen, ...) der eingesetzten Hochsprache (C#, Java, C++, ...) abbilden kann, jedoch dank ihres weniger komplexen Aufbaus schneller in Maschinen-Code übersetzbar ist. IL-Code enthält zusätzlich zu den Programmweisungen Metainformationen (z.B. Vererbungsstrukturen). Dieses Vorgehen ermöglicht einem Entwickler, der die .Net-Technologie nutzen möchte, die fast freie Wahl der Programmiersprache.

⁹⁸ Bei .Net dient die **CLR** (Common Language Runtime) einerseits als Abstraktionsschicht zum Betriebssystem und zur vorhandenen Hardware, andererseits stellt sie die Unabhängigkeit von der verwendeten Programmiersprache sicher, dieses Vorgehen sichert Plattformunabhängigkeit.

⁹⁹ **JIT** Abk. für „Just in time“. Hauptvorteil der JIT-Compilation ist, dass der JIT-Compiler auf dem Zielsystem in der Lage ist Optimierungen entsprechend dem verwendeten System vorzunehmen.

2.3 Gauß-Krüger-Koordinaten

Mit Gauß-Krüger-Koordinaten¹⁰⁰ lassen sich Punkte auf der Erdoberfläche genau lokalisieren. Diese so genannten Lage-Koordinaten werden in IMMOMENT benötigt, um zu einer Hausanschrift deren lagegenaue Position in Landkarten anzeigen zu können. Die Firma IMMO-DATA AG wird beispielsweise durch die folgenden Lage-Koordinaten lokalisiert: Hochwert = 2578367 und Rechtswert = 5704263.

Ein Partner liefert IMMOMENT in seiner Datenbank zu allen ihm bekannten Hausanschriften das entsprechende Koordinatenpaar (in drei Qualitätsausprägungen - s. Kapitel zehn), mit deren Hilfe bei Aufruf „der Dienst Objektlage“ (s. Kapitel zehn) die Bilder vom entsprechenden Partner angefordert werden können. Wie IMMOMENT aufgrund der Adresse die Lage-Koordinaten ermittelt, wird in dem Kapitel sechs „Daten- und Kommunikationsmodell“ erläutert.

Zur Lokalisierung eines Punktes auf der Erdoberfläche stehen viele Koordinaten-Systeme zur Auswahl. Das Gauß-Krüger-Koordinaten-System hat den Vorteil, dass es in metrischen Werten angegeben ist, das erleichtert die Entfernungsberechnung zweier Punkte, die im folgenden Unterkapitel „Datensuche durch Lage-Koordinaten“ beschrieben wird. Eine weitere Voraussetzung, die für die Gauß-Krüger-Koordinaten sprach, war dass die beteiligten Partner, die durch Lage-Koordinaten Daten anzeigen, ebenfalls mit dem Gauß-Krüger-Koordinaten-System arbeiten.

¹⁰⁰ Eine Homepage der Universität Mannheim erläutert die **Gauß-Krüger-Koordinaten** folgendermaßen: „Das deutsche Gauß-Krüger System teilt die Fläche der Bundesrepublik in Meridianstreifensysteme ein, die um die Zentralmeridiane 6°, 9°, 12° und 15° eine Ausdehnung von jeweils 1,5° (entspricht etwa 100 km) haben. An den Überlappungsstellen der einzelnen Meridianstreifensysteme werden die Gauß-Krüger-Koordinaten, deren Fehler (Längenverzerrung) sich mit zunehmendem Abstand von Zentralmeridian vergrößert, aufeinander abgestimmt.

Gauß-Krüger-Koordinaten werden in metrischen Werten angegeben (Kilometer oder Meter). Der Rechtswert gibt die Lage eines Punktes in West-Ost-Richtung an und bezieht sich auf den Zentralmeridian des jeweiligen Meridianstreifensystems. Der Zentralmeridian entspricht dem Wert 500.000 Meter oder 500 Kilometer, damit für westlich des Zentralmeridians gelegene Punkte keine negativen Werte entstehen. Die erste Ziffer des Rechtswertes ergibt mit drei multipliziert den Zentralmeridian des Meridianstreifensystems. Der Hochwert gibt die Lage eines Punktes in Nord-Süd-Richtung an und entspricht der Entfernung eines Punktes vom Äquator.“ [24]

Das „Informationsblatt über Bezugssysteme, Koordinaten und Höhen der Landesvermessung in Nordrhein-Westfalen“ informiert darüber hinaus noch über die geschichtliche Entwicklung des Koordinatensystems: „Geodätische Grundlage der Lagefestpunkte in der Bundesrepublik Deutschland ist das Deutsche Hauptdreiecksnetz (DHDN). Es besteht in Nordrhein-Westfalen aus dem 1895 fertig gestellten Schreiberschen Block der Preußischen Landesaufnahme (Pr.LA.). Die Koordinaten der Festpunkte sind auf das Erdellipsoid von Bessel bezogen; Zentralpunkt ist der Hauptdreieckspunkt Rauenberg in Berlin. Dieses Bezugssystem wird als Potsdam Datum (Zentralpunkt Rauenberg) bezeichnet. Seit 1927 werden für die Lagefestpunkte rechtwinklige Gauß-Krüger-Koordinaten (y, x) in 3° breiten Meridianstreifen bestimmt. Koordinaten des Systems Pr.LA. sind Ausgangswerte für Vermessungen im Liegenschaftskataster, für topographische Karten und geographische Informationssysteme (GIS), sie werden mit dem Lagestatus 101 bezeichnet.“ [25]

2.4 Datensuche durch Lage-Koordinaten

Derzeit ermöglichen zwei Dienste dem Nutzer die Anwahl von Firmen¹⁰¹-Informationen in einem auswählbaren Umkreis zur eingegebenen Adresse, diese beiden Dienste sind die „Infrastruktur“ und das „Branchenumfeld“.

Nach Anwahl der Dienste kann bestimmt werden, in welchem Umkreis (z.B. 500 m) um das Objekt herum Firmen-Daten angezeigt werden sollen.

Die Suche der Daten wird in zwei Schritten durchgeführt:

Im **ersten Schritt** werden nur die Firmen ermittelt, die sich im Quadrat um das Objekt herum befinden. Im vorherigen Unterkapitel wurden bereits die Gauß-Krüger-Koordinaten vorgestellt, sie werden mit metrischen Werten angegeben, aus diesem Grunde liegen die gesuchten Firmen im Quadrat mit der Seitenlänge $2 * \text{Radius}$. Da die entsprechenden Felder der Tabelle mit einem Index versehen wurden, wird diese Prüfung sehr schnell durchgeführt. Die Abfrage ist direkt als SQL-Befehl möglich, da sowohl von allen Objekten als auch von den Firmen die Lage als Gauß-Krüger-Koordinatenpaar vorliegt. Folgender SQL-Befehl zeigt den Aufbau für die Koordinaten $X=2001000$ und $Y=5001000$ und den Radius 500 m:

```
SELECT HAUS_ID from haus_kern where HAUS_Gewerbeinfo > 0 and  
      HAUS_Koo_Rechts > 2000000 and HAUS_Koo_Rechts < 2002000 and  
      HAUS_Koo_Hoch > 5000000 and HAUS_Koo_Hoch < 5002000
```

Im **zweiten Schritt** wird nun geprüft, ob jede gefundene Firma auch im Inkreis des Quadrates liegt. Diese Prüfung ist zeitintensiver, da jede bereits gefundene Firma des ersten Schrittes nun einzeln durch die Punkt-Abstands-Formel¹⁰² geprüft werden muss.

Wurde nach allen Prüfungen kein Eintrag gefunden, der die Kriterien erfüllt, wird eine Textbox mit dem folgendem Hinweis gefüllt: „Keine Einträge im ausgewählten Ausschnitt vorhanden!“. Wurden nach beiden Schritten Firmen für den entsprechenden Radius um das vorgegebene Objekt gefunden, so müssen nun noch aus der Tabelle „firmen“ der Name der Firma nachgelesen werden bzw. weitere Informationen dieser Tabelle die in dem speziellen Dienst angezeigt werden sollen.

¹⁰¹ Objekte (Immobilien) mit **Firmen** werden daran erkannt, dass das Attribut HAUS_Gewerbeinfo der Tabelle „HAUS_KERN“ einen Wert größer „0“ enthält.

¹⁰² **Punkt-Abstands-Formel:** Der Abstand zweier Punkte in einem Koordinatensystem wird wie folgt berechnet: $\text{Abstand} = \sqrt{(x_{p1} - x_{p2})^2 + (y_{p1} - y_{p2})^2}$ x_{p1} (x_{p2}) steht für den X-Koordinatenabschnitt des ersten (zweiten) Punktes und y_{p1} (y_{p2}) für den Y-Koordinatenabschnitt des ersten (zweiten) Punktes. Wurzel der Summe der Quadrate der Abstände der Koordinatenabschnitte.

2.5 Datenschutz¹⁰³

Mit Datenschutz ist der Schutz der Persönlichkeitsrechte - der Privatsphäre - von Personen beim Umgang mit ihren Daten gemeint. Das Datenschutzrecht versucht dabei, zwischen den Interessen der Datenverarbeiter und den Persönlichkeitsrechten der Betroffenen einen Ausgleich herbeizuführen.

Insbesondere ist das Persönlichkeitsrecht (bzw. die Anonymität) des Einzelnen bei personenbezogenen Daten¹⁰⁴ zu wahren. Unter personenbezogenen Daten sind Informationen zu verstehen, die eindeutig einer Person zugeordnet werden können bzw. die explizit genutzt werden können, um über die Identität einer bestimmten Person etwas zu erfahren.

Im Dienst Wohnumfeld (s. „8 Wohnumfelddaten“) werden viele Daten über Haushalte angezeigt, die aus Datenschutz-Sicht kritisch wirken. Die Erläuterung des Partners, der für IMMOMENT diese Daten liefert, soll darstellen, wie durch Segmentbildung die Persönlichkeitsrechte des Einzelnen gewahrt bleiben; dieses Vorgehen wurde auch vom Bundesamt für Datenschutz zugelassen: „... Diese Informationen liegen grundsätzlich für sämtliche (rund 34. Mio.) Haushalte Deutschlands vor und werden für die rund 17 Mio. Häuser ausgewiesen. Für die Analyse werden – aus Gründen des Datenschutzes – mehrere zu einem Wohnumfeld gehörende Häuser zu einem ‚virtuellen‘ microgeographischen Segment gebündelt, das mindestens fünf, durchschnittlich acht Haushalte umfasst. Dabei erfolgt eine ‚Begehung der Straßen im Computer‘. Zunächst werden die Häuser danach geprüft, ob wenigstens fünf Haushalte darin vorkommen. Große Häuser bilden also eigene Segmente. Wo dies nicht gegeben ist, werden in jeder Straße strukturähnliche Häuser zu Segmenten zusammengefasst, in denen zusammen mindestens 5 Haushalte vorkommen.“

¹⁰³ **Datenschutz** ist der normative Auftrag an juristische Personen des privaten und öffentlichen Rechts, das informationelle Selbstbestimmungsrecht des Einzelnen vor unzulässiger Beeinträchtigung zu schützen. Die nicht-automatisierte und automatisierte Erhebung, Speicherung, Verarbeitung und Übermittlung personenbezogener Daten ist nur zulässig, wenn dies durch ein Gesetz oder andere Rechtsvorschrift ausdrücklich bestimmt.

¹⁰⁴ „**Personenbezogene Daten** im Sinne des Bundesdatenschutzgesetzes (BDSG) und des Sozialgesetzbuches (SGB I, X) sind Einzelangaben über persönliche und sachliche Verhältnisse einer bestimmten oder bestimmbarer natürlichen Person. Dies können je nach Vertragsart neben Name, Anschrift, Geburtsdatum, Beruf und Bankverbindung auch alle sensitiven besonderen Daten sein, wie z.B. Gesundheit, Einkommensverhältnisse und finanzielle Bonität. All diese Daten unterliegen gemäß BDSG einem besonderen Schutz.“ [26]

3 Die Entwurfsphase

Die Planungsphase ergab folgenden Workflow¹⁰⁵ (im Grunde das EVA-Prinzip¹⁰⁶):

- Ein Nutzer gibt über einen Internetbrowser nach der erfolgreichen Anmeldung (login) eine Adresse ein.
- Nach der erfolgreichen Adress-Prüfung (Abbildung 1) erhält der Nutzer Daten.
- Der Nutzer verlässt IMMOMENT (logout) oder gibt eine neue Adresse ein.

Das Pflichtenheft als Ergebnisdokument der Definitionsphase zeigte auf, dass vor der Implementierung mehrere funktionale und nicht-funktionale Anforderungen¹⁰⁷ bedacht werden müssen:

1. Die Daten müssen dem Nutzer über einen Internetbrowser zugänglich sein und auch in diesem angezeigt werden können, möglichst ohne dass sich der Nutzer noch spezielle Software installieren muss. Auch die Wahl des Internetbrowsers (Produkt und Version) sollte dem Kunden möglichst freigestellt sein.
2. IMMOMENT sollte zwei verschiedene Arten von Nutzern unterscheiden, neben den „normalen“ Nutzern sollte es möglich sein, Interessenten bzw. Testuser für einen Zeitraum oder eine bestimmte Anzahl von Logins den Zugang zu gewähren. Die Authentifizierung des Nutzers muss folglich mehrere Prüfungen enthalten (Zeitraum der Anmeldeerlaubnis, möglicherweise IP-Adressen¹⁰⁸-Überprüfung, Beschränkung der Login-Anzahl).
3. Zusätzlich müssen auch Demo-User einen ungeprüften Zugang erhalten. Der Demo-User wird in einem allgemeinen Demobereich geführt, in dem nur wenige

¹⁰⁵ Ein **Workflow** ist eine Kette von Arbeitsschritten, die immer wieder durchlaufen werden.

¹⁰⁶ **EVA-Prinzip** steht für Eingabe-Verarbeitung-Ausgabe, Daten werden z.B. über die Tastatur eingegeben, vom Computer verarbeitet und anschließend z.B. vom Bildschirm ausgegeben.

¹⁰⁷ „**Funktionale Anforderungen** beziehen sich auf funktionale Aspekte eines Systems u. beantworten die Frage, was ein System tut. Demgegenüber betreffen **nicht-funktionale Anforderungen** zusätzliche Eigenschaften, die die Systemfunktionen aufweisen sollen (Qualitätsattribute), Vorgaben für die Durchführung der Systemerstellung sowie Einführung, Betreuung und Betrieb des Systems. In der Regel haben die funktionalen Anforderungen ein wesentlich größeres Gewicht als die nicht-funktionalen.“ [5, S. 6]

¹⁰⁸ „Eine **IP-Adresse** (IP = Internet Protocol) ist ein eindeutiges Identifikationsmerkmal, das den Standort eines Computers im Internet definiert. Sie gibt Aufschluss über den verwendeten Provider und den Standort des Servers. IPs sind 32-Bit-Adressen, die aus vier durch Punkte (.) getrennte Byte-Blöcke (Nummern zwischen 1 und 255) bestehen. Jede IP-Adresse ist einem bestimmten Hostnamen zugeordnet, da es für den Benutzer in der Regel einfacher ist, sich einen Namen zu merken als eine solche Nummer.“ [27] Z.B. - die IP-Adresse: 212.184.80.190 steht für den Hostnamen: www.microsoft.de

fest vorgegebene Adressen (keine freie Adresseingabe möglich) anwählbar sind. In dem Demobereich werden keine Partner-Server aufgerufen, die angezeigten Daten sind in HTML-Seiten fest vorgegeben. Es muss ebenfalls möglich sein, sich diesen Demobereich (z.B. per FTP¹⁰⁹) herunterladen (download) zu können, um Offline die vorgegebenen Adressen aufrufen zu können.

4. IMMOMENT ist eine „pay-per-Click“ oder „pay-on-demand“ Anwendung, aus diesem Grunde darf der Nutzer nicht sofort alle verfügbaren Daten für eine vorgegebene Adresse angezeigt bekommen, sondern diese müssen einzeln angefordert werden (Dienstaufrufe).
5. Es kann vorkommen, dass nicht alle Dienste zu einer Adresse vorliegen, somit sollte dem Nutzer bereits nach der Adressprüfung angezeigt werden, welche aufrufbar sind und welche nicht zur Verfügung stehen (z.B. durch grau gefärbte Button im Hauptmenü s. Abbildung 1).
6. Jedem Nutzer sind Kompetenzen zuzuordnen, die regeln, welche Dienste aufgerufen werden dürfen. Dienste, die der Nutzer nicht aufrufen darf, müssen kenntlich gemacht werden (z.B. durch blau gefärbte Button im Hauptmenü).
7. Textdaten werden in der relationalen Datenbank SQL-Server¹¹⁰ verwaltet.
8. Neben dem dialoggesteuerten Internetbrowser ist die Anforderung von Daten über Web-Services zu unterstützen, wodurch die Daten lokal auf dem Client-Rechner des Nutzers „heruntergeladen“ werden. Es muss auch bei der Datenübermittlung über Web-Services berücksichtigt werden, dass nicht automatisch alle lieferbaren Daten zurückgegeben werden.
9. Jeder Dienstaufruf, ob über den Internetbrowser oder einen Web-Service, muss einzeln nachvollziehbar sein, damit dem Nutzer eine Rechnung ausgestellt werden kann. Es muss folglich ein so genanntes „Billing-System“ umgesetzt werden.

¹⁰⁹ „FTP (File Transfer Protocol) ist ein standardisierter Satz von einfachen Befehlen, die das Transferieren von Dateien von einem Internet-Computer auf einen anderen ermöglichen. Die Funktionalität von FTP ist in jedem WWW-Client integriert. Besondere Bedeutung hat dieser Internet-Dienst aufgrund von anonymem, das heißt öffentlichem FTP-Zugriff auf eine Vielzahl von FTP-Servern, auf deren Massenspeichern Megabytes an [...] Software von den Server-Betreibern allen Internet-Teilnehmern zur Verfügung gestellt werden. Ein URL für FTP wäre zum Beispiel <ftp://ftp.wu-wien.ac.at/pub/WWW/mac/>.“ [8]

¹¹⁰ SQL (Structured Query Language) ist eine Definitions- und Abfragesprache für relationale Datenbanken. Trotz Normung (SQL92, SQL99) haben sich verschiedenste Dialekte entwickelt – **SQL-Server** ist eine relationale Datenbank, die von der Firma Microsoft angeboten wird. Es handelt sich aus Sicht der Klassifizierung von Datenbanksystemen nach Art der Anwendung um ein Datenverarbeitungssystem. SQL gehört zu den Sprachen der vierten Generation, der Programmierer löst ein Problem, indem er beschreibt, was der Computer tun soll, der Weg dahin – das „wie“ – findet die Sprache selbst.

10. Die Daten des Billing-Systems müssen dem Nutzer als Rechnung zugeschickt werden. Die Rechnungserstellung muss ebenfalls über einen Internetbrowser aufrufbar und auch für Nicht-Programmierer „einfach“ anzuwenden sein.
11. Die Web-Applikation IMMOMENT sollte unabhängig vom Provider¹¹¹ (bzw. Hosting¹¹²) „überall“ einsatzfähig sein, somit muss die Administration bestimmter Daten (Nutzererfassung, Grunddaten wie Kostenvorgabe für die Dienste und einige mehr) ebenfalls über einen Internetbrowser möglich sein.
12. Die Anwendung wird in einer Webfarm¹¹³ eingesetzt, dabei werden die einzelnen Webserver per „load balancing“¹¹⁴ angesteuert. Ein Nutzer kann sich von einer Anfrage zur nächsten auf einem anderen Server „befinden“, somit ist bei der Implementierung zu bedenken, dass eine spezielle Session-Verwaltung nötig ist.
13. IMMOMENT „lebt“ durch seine Partner, denn viele Daten werden erst beim Dienstaufwurf des Nutzers von Partner-Servern angefordert. Es muss berücksichtigt werden, dass jederzeit ein weiterer Anbieter hinzukommen kann.
14. Die Server IP-Adressen der Partner können sich, über einen längeren Zeitraum betrachtet, ändern, somit muss bedacht werden, dass bei einer IP-Adressen-Änderung keine Anpassung im Quellcode nötig wird. Eine Steuerung der Partner-Server über eine Datenbank muss eingeplant werden.
15. Die Konfiguration der Firewall des Providers muss berücksichtigt werden, sollte diese nicht in der Lage sein URL¹¹⁵-Namen (über einen DNS-Server) aufzulösen, also Zugänge rein nach IP-Adressen zu gewähren, müssen die Partnerserver entsprechend aufrufbar sein. Aus diesem Grunde müssen genaue Vorgaben zwischen den Partnern und dem Provider abgesprochen werden.

Aus diesen Punkten ergibt sich, dass bereits vor der Anmeldung IMMOMENT grundsätzlich zwei Zugänge erlauben muss:

- Zugang per Internetbrowser und
- per Web-Service

¹¹¹ **Provider** ist das Unternehmen, das gegen Gebühr den Zugang zum Internet ermöglicht.

¹¹² **Hosting**, auch „houseing“ genannt, bezeichnet die Möglichkeit, einen eigenen Rechner bei einem Provider aufzustellen. Dies spart die Gebühren für die sonst nötige Standleitung.

¹¹³ Bei einer **Webfarm** wird statt eines Einzigen eine Reihe von Webservern verwendet, die alle die gleichen Aufgaben übernehmen und sich gegenüber dem Besucher als ein Websystem präsentieren.

¹¹⁴ **Load Balancing** Systeme verteilen die Last zwischen Geräten (Servern) in einer Web-Umgebung

¹¹⁵ „**URL** bedeutet Uniform Resource Locator, beispielsweise <http://www.commando.de/verz/dat.ext> - die eindeutige Adresse eines Internet-Rechners bzw. einer bestimmten Information darauf. Der Inhalt und das Übertragungsprotokoll der URL wird durch den Teil vor dem Doppelpunkt bestimmt; [http:](http://) steht für WWW-Seiten, [ftp:](ftp://) für die entsprechenden Dienste; hinter [news:](mailto:) oder [newsrc:](mailto:) verbergen sich Newsgroups und hinter <mailto:> steht eine eMail-Adresse.“ [28]

Durch die Anmeldung müssen verschiedene Nutzer erkannt werden:

- normaler Nutzer, der zu einer Adresse Daten anfordert,
- Interessenten bzw. Testuser, die zu einer Adresse Daten anfordern,
- Demo-User, die zu fest vorgegebenen Adressen Daten online anfordern,
- Demo-User, die sich den Demobereich herunterladen (download),
- Administratoren, die wieder unterteilt werden müssen in:
 - i. Administratoren, die Daten manipulieren dürfen (z.B. Nutzerdaten),
 - ii. Administratoren, die nur Daten ausgeben dürfen (Rechnungen erstellen).

Die verschiedenen Arten von Nutzern haben zur Folge, dass nach der Anmeldung unterschiedliche Startseiten (Adresseingabe, Demobereich oder Administrationsseiten) angeboten werden müssen.

Ein Status regelt zusätzlich Zugriffsrechte:

- Bei den Administratoren regelt der Status, ob Daten manipuliert werden dürfen.
- Bei den Nutzern wird zwischen „normalen“ oder Interessenten bzw. Testusern unterschieden.

Neben den unterschiedlichen Startseiten und den Zugriffsrechten, wird durch Kompetenzen gesteuert, welche Dienste ein „normaler“ Nutzer aufrufen darf.

4 Partner



Abbildung 2 – Web-Ansicht der Partner der IMMO-CHECK GmbH

Terra Map Server GmbH

Hostet IMMOMNENT über eine Web-Farm und beliefert die Dienste (Bilddaten):
Stadtpläne, Lageplan, Topographische Karten und Luftbilder

Microm Marketing und Consult GmbH

Beliefert die Dienste:
Wohnumfeld, Firmendaten, Risiken, Infrastruktur und Branchenumfeld

Bulwien AG

Beliefert den Dienst: Marktdaten

SEC – Software Entwicklung und Concept GmbH

Beliefert die Dienste: Bodenrichtwerte und Miet-/Preisspiegel

Abis AG

Führt die Adressprüfung durch

Tele-Info Digital Publishing AG

Beliefert die Dienste (Straßenbilder):
City-View und Sity-Server

Schufa Holding AG

Schufa steht für die „Schutzgemeinschaft für allgemeine Kreditsicherung“
Creditreform e.V.

CEG – Creditreform Experian GmbH

IMMO-DATA AG

5 Prinzip des Datenaustauschs

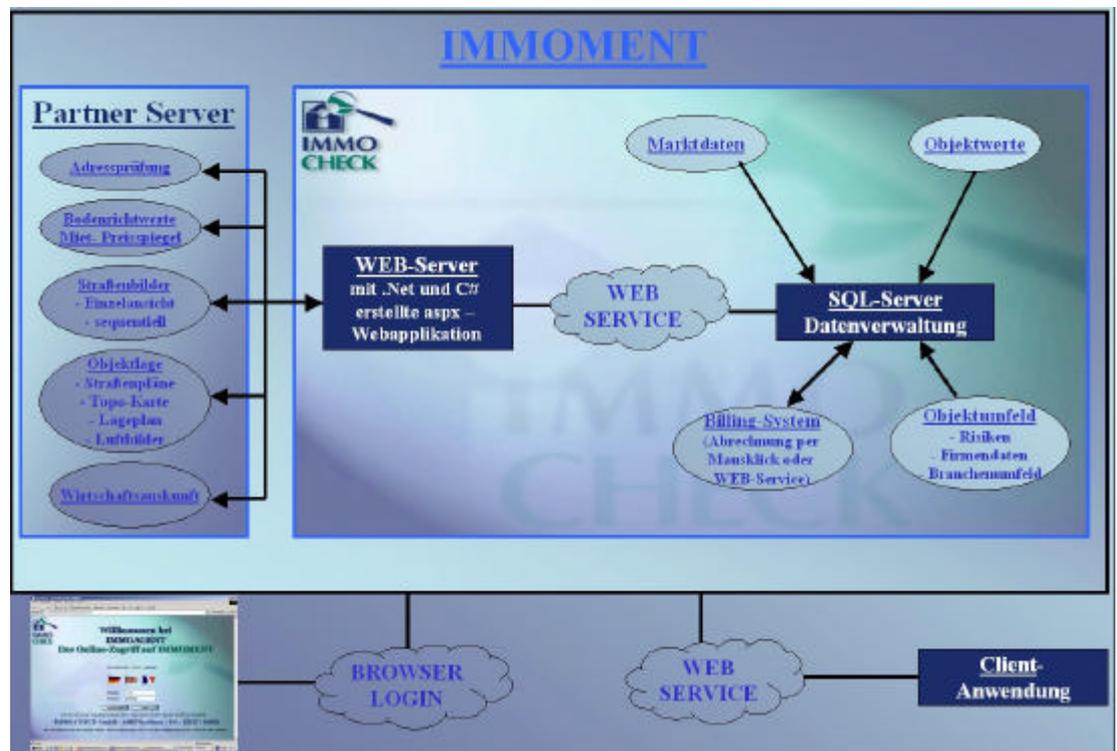


Abbildung 3 – Prinzip des Datenaustauschs

Es wurde bereits erwähnt, dass IMMOMENT ein „Informationssammler“ ist, das bedeutet, dass durch die Web-Applikation viele Text- und Bildinformationen von verschiedenen Partnern gebündelt dem Nutzer zur Verfügung gestellt werden.

Dieses Kapitel zeigt den schematischen Überblick von IMMOMENT. „Abbildung 3“ veranschaulicht, dass die Web-Applikation IMMOMENT aus mehreren Teilen besteht.

Der rechte Teil (IMMO-CHECK) stellt mit dem „Web-Server“, der auf den „SQL-Server“ über Web-Services zugreift, die grundsätzliche Anwendung dar. Dabei verwaltet der SQL-Server neben anzuzeigenden Daten (Marktdaten, Objektwert und Objektumfeld), auf die nur lesend zugegriffen wird, auch Verwaltungsdaten z.B. des Billing-Systems, die zur Rechnungserstellung dienen und somit auch in die entsprechenden Datenbanktabellen geschrieben werden müssen.

Es liegen aber nicht alle Daten, die IMMOMENT anzeigen kann, auf dem eigenen SQL-Server vor und auch die Adressprüfung wird über einen Partner-Server durchgeführt.

Der linke Teil der Abbildung zeigt, dass durch bestimmte Dienstaufrufe (Bodenrichtwerte, Straßenbilder, Objektlage und Wirtschaftsauskunft) des Nutzers Daten von den Partner-Servern angefordert werden. Die Partner stellen ihre Daten, entweder durch Web-Services oder durch HTTP-Requests, zur Verfügung.

Der untere Teil der Abbildung stellt den Zugriff zu IMMOMENT dar. Die Daten von IMMOMENT können von außen auf zwei Wegen angefordert werden, entweder über den dialogorientierten Weg über einen Internetbrowser (Browser-Login) oder über Web-Services, wodurch die angeforderten Daten in lokale Systeme des Client-Rechners integriert werden können.

6 Daten- und Kommunikationsmodell

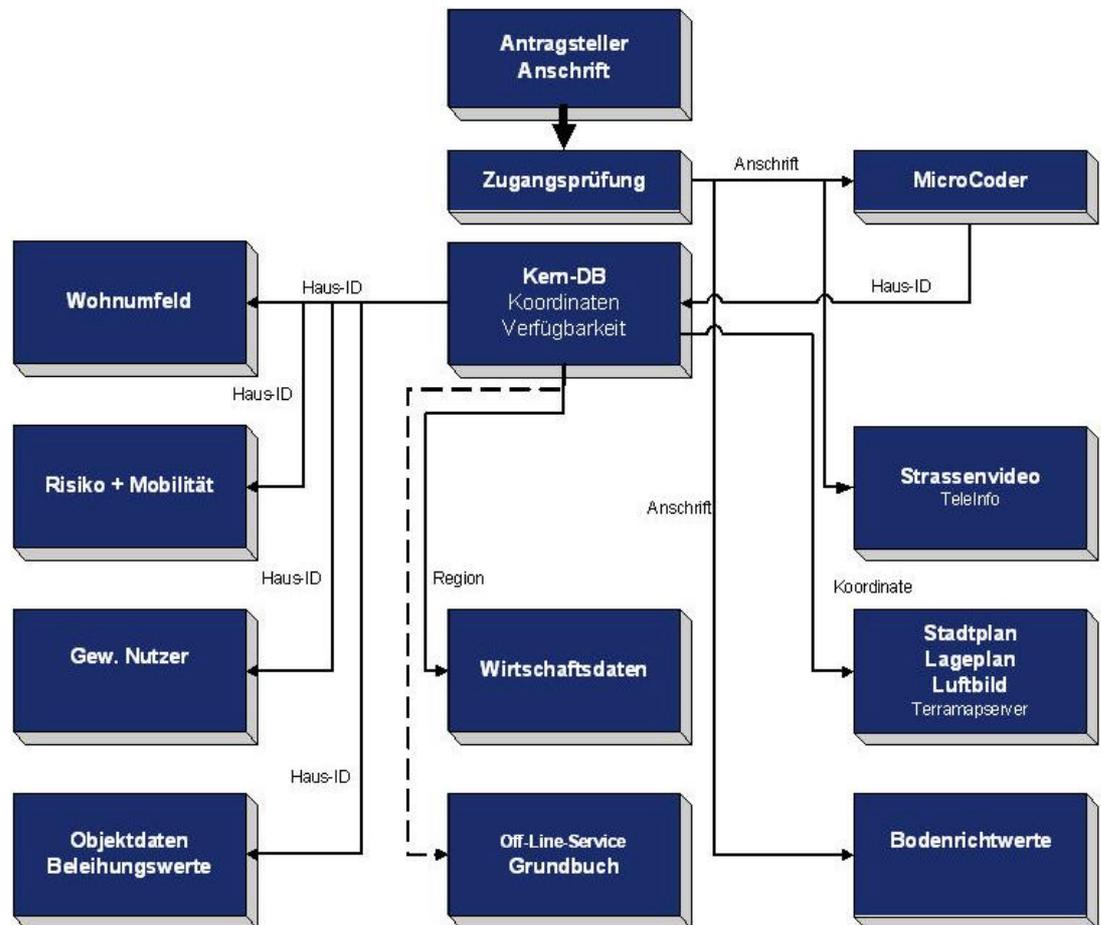


Abbildung 4 – Daten- und Kommunikationsmodell

Das vorliegende Kapitel hebt hervor, dass bestimmte Schlüssel-Daten benötigt werden, damit dem Nutzer die verfügbaren Informationen angezeigt werden können.

„Abbildung 4“ zeigt, dass der Aufruf der Daten durch drei Schlüssel-Attribute erfolgt:

- Anschrift (PLZ, Ort und Straße mit bzw. ohne Hausnummer)
- Haus-ID
- Lage-Koordinaten (s. „2.3 Gauß-Krüger-Koordinaten“)

Nach der Adressprüfung können ohne weitere Datenbankabfrage mit der Anschrift (ohne Hausnummer) „Bodenrichtwerte“, bzw. „Miet-/Preisspiegel-“ Daten ermittelt werden. Alle weiteren Dienste benötigen die Haus-ID direkt, um aus weiteren Tabellen die geforderten Informationen anzeigen zu können (s. „Abbildung 8 – Datenbankschema der Wohnumfelddaten“) oder indirekt um aus der so genannten Kern-DB (Tabelle Haus_Kern) die Lage-Koordinaten für das Objekt ermitteln zu können.

Mit den Kage-Koordinaten können dann bei verschiedenen Partnern Bilddaten angefordert werden (s. „10 Der Dienst Objektlage“).

Die Adressprüfung durch den „MicroCoder“, die ein Partner-Server durchführt, wird im Folgenden genauer dargestellt. Für die Adressprüfung werden wenigstens folgende Werte benötigt:

Die Postleitzahl (PLZ) und / oder der Ort und die Straße inkl. der Hausnummer. Dabei prüfen reguläre Ausdrücke¹¹⁶, ob die Eingabe den Anforderungen entspricht. Wurde z.B. die Hausnummer vergessen, erscheint der Hinweis: „Die Straße muss wenigstens zwei Zeichen lang sein und eine Hausnummer enthalten.“

Die PLZ ist zwar keine Pflichteingabe, da einige Orte in Deutschland aber mehrfach vorkommen (z.B. Neustadt) bzw. innerhalb eines Ortes eine Straße mehrfach vorkommen kann, ist es sinnvoll, auch die PLZ einzugeben. In den meisten Fällen kann das System die korrekte Adresse aber auch ohne PLZ ermitteln. Sollte die Eingabe eine nicht eindeutige Adresse ergeben, so wird vom Partner eine Liste der möglichen Adressen zurückgegeben, welche dem Nutzer als Auswahlliste oder Listbox angezeigt wird. Folglich muss nun die richtige Adresse ausgewählt werden (s. „Abbildung 5“).

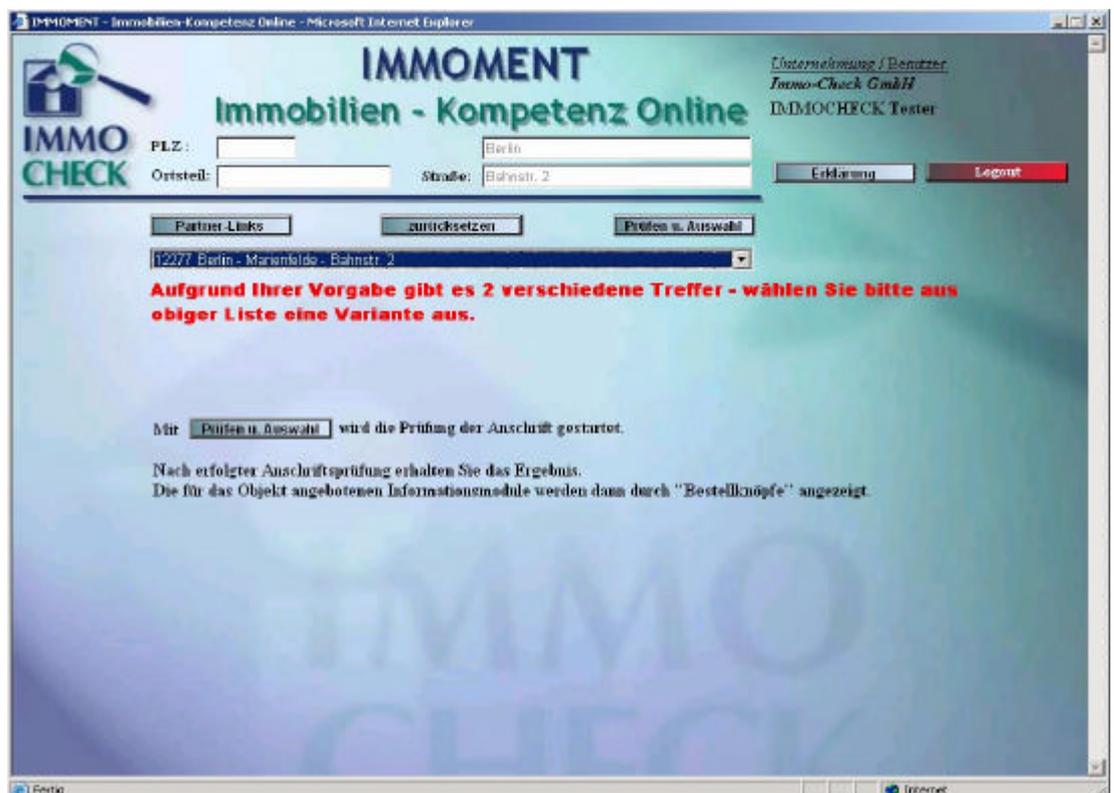


Abbildung 5 – Web-Ansicht – Adresseingabe: Nicht eindeutige Adresse

¹¹⁶ **Reguläre Ausdrücke** stellen Suchmuster dar, sie lassen sich auf Strings (Texte) anwenden und für sie ist entscheidbar, ob sie auf den String passen (*match*) oder nicht. Reguläre Ausdrücke können z.B. folgende Problemstellung analysieren: Sind alle Wörter (möglicherweise getrennt durch Bindestriche) wenigstens zwei Zeichen lang – oder – steht am Ende der Zeile eine Zahl, der höchstens ein Buchstaben oder Leerzeichen folgt (zum Erkennen der Hausnummer).

Der Partner prüft, ob die eingegebene Adresse korrekt ist, d.h. es wird überprüft, ob die Kombination von PLZ, Ort und Straße in einem Straßenverzeichnis aller Straßen Deutschlands vorhanden ist. Dabei wird aber eine intelligente Prüfung durchgeführt, die nicht nur 1:1 die Werte abgleicht, sondern es werden durch bestimmte Verfahren Ähnlichkeiten entdeckt. Durch diese Verfahren können selbst Buchstabenvertauschungen oder fehlende Teile des Wortes so ergänzt werden, dass die Zuordnung zu einer Straße möglich ist. Ist die falsch geschriebene Variante sehr nah an einer entdeckten Straße, so wird diese übernommen und der Nutzer bekommt in einer neuen Bildschirm-Maske die gefundene Straße angezeigt, die Unterschiede in der Schreibweise werden angezeigt.

Zusätzlich zur korrigierten Adresse übermittelt der Partner den so genannten „return-code“ – eine vierstellige Ziffer, die zurückliefert, ob die Eingabe korrekt war (return-code = „1111“), oder falls nicht, welche Elemente falsch waren und welcher Hinweistext bzw. welche Warnfarbe angezeigt werden muss.

Das Hauptmenü („Abbildung 1“) wird direkt nach der Adressprüfung angezeigt, in dem Beispiel auf Seite fünf ist zu erkennen, dass einzelne Teile der Adresse beurteilt und durch Farben gekennzeichnet werden. Dabei sind die Farben entsprechend dem Warnsystem der Ampelfarben gewählt worden, sie werden durch einen Text erläutert:

1. *Grün* bedeutet, dass der entsprechende Teil korrekt ist (in dem Beispiel sind die PLZ und der Ort richtig geschrieben worden. Der Hinweistext erläutert: „Die entsprechend gekennzeichneten Elemente der Anschrift sind korrekt.“
2. *Gelb* bedeutet, dass die Eingabe nicht in Ordnung war, dass aber mit sehr hoher Wahrscheinlichkeit die richtige Straße gefunden werden konnte. In der „Abbildung 1“ wurde die Straße mit „Burgerstr. 21“ eingegeben, das System hat die „Burgstr. 21“ ermittelt und keine andere ähnliche Straße entdeckt. Die Farbe und der entsprechende Hinweis-Text sollen den Nutzer auffordern, die gefundene Adresse noch einmal zu überprüfen: „Das markierte Anschriftelement ist nicht korrekt. Aus dem Kontext konnte jedoch mit großer Sicherheit die richtige Schreibweise ermittelt werden. Die Zuordnung ist deshalb mit großer Wahrscheinlichkeit zutreffend. Sind jedoch zwei Elemente rot markiert, erhält das möglicherweise korrekte Element der Anschrift statt einer grünen eine gelbe Kennung.“
3. *Rot* hingegen lässt keine klare Aussage zu, da mehrere Eingaben falsch sind. Folgendes gibt der Hinweis-Text an: „Mindestens zwei Elemente der Anschrift sind nicht korrekt. Aus dem Kontext konnte zwar eine Verknüpfung mit gewisser

Wahrscheinlichkeit konstruiert werden, es wird jedoch dringend empfohlen, die Ergebnisse mit Vorbehalt zu nutzen und zunächst die Anschrift zu überprüfen.“

Der Partner prüft die Hausnummer zwar nicht, ermittelt aber neben der richtig geschriebenen Adresse und dem „return-code“ die so genannte Haus-ID. Die Haus-ID besteht aus einer 18 bzw. 19-stelligen Zeichenfolge, deren Aufbau das Bundesland, den Landkreis, die Gemeinde, die Straße und Hausnummer enthält. Zur Erstellung der Haus-ID wird folglich die Hausnummer benötigt (s. Anhang A „4. Produkt-Funktionalität“ Abschnitt b).

Erst in einem zweiten Schritt wird geprüft, ob die ermittelte Haus-ID in der Kern-DB vorhanden ist. Ist sie es nicht, so ist die Hausnummer in dieser Straße nicht bekannt und es erscheint im roten Feld der folgende Warnhinweis: „Die Hausnummer konnte nicht gefunden werden - prüfen Sie diese bitte. Falls es sich um einen Neubau handelt, ist dieser möglicherweise noch nicht in unserem Verzeichnis eingetragen. Zu dem Straßenabschnitt können wir Ihnen Informationen anbieten.“ (s. „Abbildung 6“).

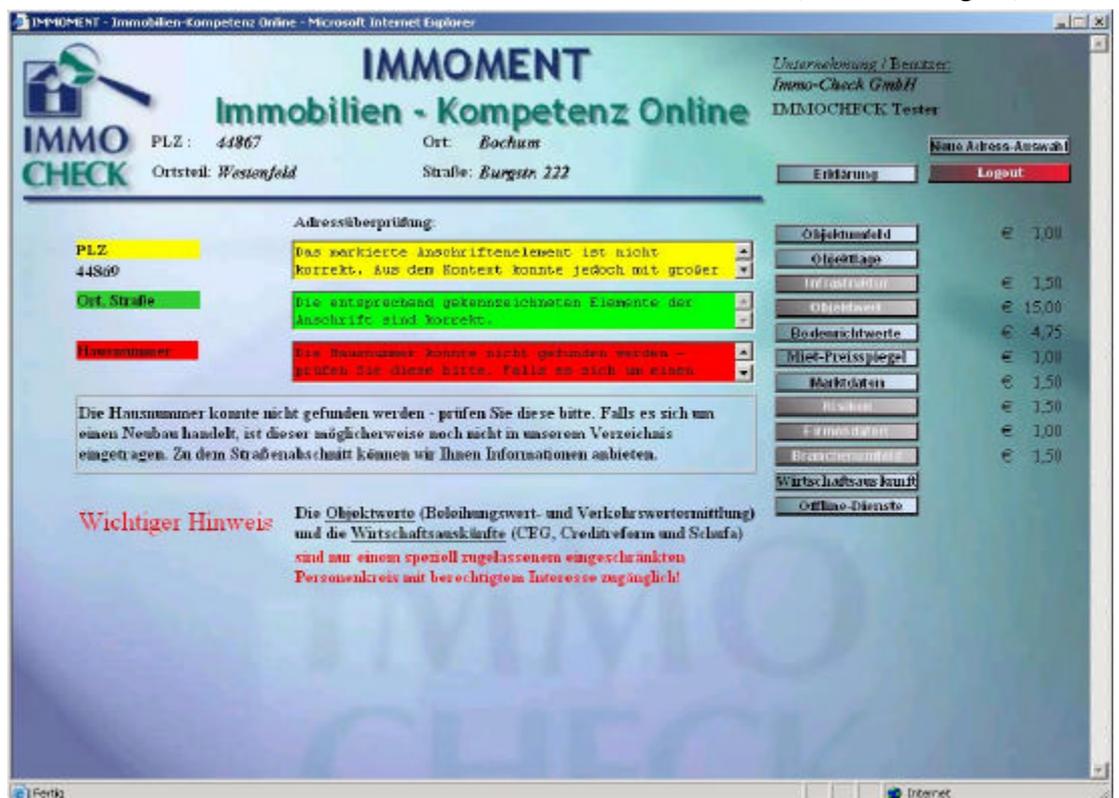


Abbildung 6 – Web-Ansicht Hauptmenü einer Straße mit unbekannter Hausnummer

Die Dienste, für die die Haus-ID unumgänglich sind, können nun nicht aufgerufen werden, das wird durch die graue Färbung der entsprechenden Button kenntlich gemacht.

7 Datenbanken, die Basis des Systems

Die Web-Applikation IMMOMENT wird, wie bereits im Kapitel „1.2 Die Art der Daten“ aufgezeigt, hauptsächlich durch ein Datenbanksystem gesteuert, das aus drei selbst erstellten (bzw. von Partnern vorgegebenen) Datenbanken besteht und aus zwei vom SQL-Server bereitgestellten Datenbanken, die vom .Net-Framework genutzt werden.

In diesem Kapitel werden die Datenbanken anhand des Datenbankschemas vorgestellt. Die Darstellungsform, die für das Datenbankschema gewählt wurde, ist von dem Microsoft-Softwareprodukt „Visio“ vorgegeben.

Visio nutzt spezielle Abkürzungen und Darstellungsformen, die Tabellen werden in zwei Spalten dargestellt:

- Die linke Spalte enthält besondere Hinweise auf die Art der Daten:
 - PK = Hauptschlüssel (primarykey)
 - FK = Fremdschlüssel (foreignkey), sie werden mit FK1, FK2,... durchgezählt
 - I = Index (Suchschlüssel), sie werden ebenfalls mit I1, I2,... durchgezählt
- Die rechte Spalte führt die Namen der Attribute auf, dabei sind optionale Attribute, die nicht gefüllt werden müssen, normal geschrieben, alle obligatorischen Attribute sind fett gedruckt.
- Zwischen den Tabellen bestehen Beziehungen, die durch einen Pfeil dargestellt werden. Dabei zeigt der Pfeil vom FK in die Richtung der Tabelle, in der dieser FK entsprechend der PK ist, allerdings liegen der Fuß- und der Zielpunkt der Pfeile nicht an dem entsprechenden Attribut an.

Die Datenbank „im_basis“ verwaltet hauptsächlich organisatorische Daten. IMMOMENT benötigt Schreibrechte auf alle Tabellen der Datenbank „im_basis“. Primär werden die Kosten-Tabellen des Billing-Systems erweitert, aber auch andere Verwaltungsdaten wie die Kunden- und Firmendaten sind in der Datenbank „im_basis“ enthalten. Im Folgenden werden die Tabellen und ihre Aufgaben erläutert.

7.1.1 Die Nutzerverwaltungstabelle im_user_daten

Die Tabelle „im_user_daten“ ist die zentrale Tabelle der Datenbank „im_basis“. Im Pflichtenheft wurden ihre einzelnen Attribute ausführlich beschrieben (s. „4. Produkt-Funktionalität – a Zugang zu IMMOMENT“), somit soll hier nur erwähnt werden, dass sie derzeit ein Attribut weniger führt. Warum das Attribut „ID_kompetenzen“ entfernt wurde, wird im Unterkapitel „7.1.5 Kompetenzen steuern die Dienst-Aufrufe“ erläutert.

7.1.2 im_button_info

Diese Tabelle enthält nicht nur die Namen der Internet-Seiten aller ansprechbaren Dienste, sondern verwaltet auch Daten, die später bei der Auflistung in den Menü-Seiten gebraucht werden, die die Dienst-Aufrufe steuern.

ID	button_text	button_name	order_in_list
Immoment1	Adressprüfung	1	-1
Immoment2	Hauptmaske	2	-2
Immoment100	Objektumfeld	100	100
Immoment3	Objektlage	200	200
Immoment201	Stadtplan	201	210
Immoment203	Topo-Karte	206	220
Immoment1100	Infrastruktur	1100	300

Tabelle 2 – Inhalt der Tabelle im_button_info

Die Ansicht „Tabelle 2“ zeigt einen Ausschnitt der Tabelle „im_button_info“. Das Attribut „ID“ listet alle Web-Seiten der Web-Applikation IMMOMENT auf (allerdings ohne die Datei-Endung „.aspx“). Das in „Abbildung 1“ dargestellte Hauptmenü spiegelt die Auflistung der Tabelle wider. Eine C#-Methode wertet auf bestimmten Menü-Seiten die Tabelle „im_button_info“ aus.

Die Button, die in der Tabelle unter „order_in_list“ einen Attribut-Wert größer als 0 führen, werden in aufsteigender Reihenfolge angezeigt. Die Menüs bestehen dabei aus zwei Ebenen, die erste zeigt nur die Dienste an, die direkt aufgerufen werden können, wie z.B. das Objektfeld. Einige Dienste sind aber erst nach einer weiteren Übersicht (Untermenü) aufrufbar (s. „Abbildung 9“), dazu gehören die Dienste der Objektlage (wie Stadtplan und Topo-Karte). Dienste, die erst durch ein Untermenü aufgerufen werden können, haben einen „button_name“-Eintrag, der sich nicht ohne Rest durch 100 dividieren lässt (Stadtplan = 201 und Topo-Karte = 206).

Diese Button werden nicht im Hauptmenü angezeigt, sondern erst dann, wenn der zugehörige Hauptmenüpunkt aufgerufen wurde (Objektlage = 200).

Die Button selbst sind JPEG-Bilder, deren Dateiname eine bestimmte Logik erfüllen muss: Sie heißen alle „buttonX.jpg“, wobei das X der Zahl des Attribut-Wertes „button_name“ entspricht.

Dieses Vorgehen macht es möglich, dass die Tabelle „im_button_info“ die Anzeige der Menüs steuert. Der Eintrag des Attributes „button_text“ wird angezeigt, wenn man mit der Maus den Button anklicken möchte, und wird später dem Kunden in der Rechnung als Text für den angewählten Dienst ausgegeben.

Der „Abbildung 7“ kann man entnehmen, dass die Tabelle „im_button_info“ eine zentrale Rolle in der Datenbank „im_basis“ einnimmt, da viele andere Tabellen das Attribut „ID“ als Fremdschlüssel benötigen.

7.1.3 im_grund_kosten und im_kosten_gruppen

Das in „Abbildung 1“ dargestellte Hauptmenü zeigt rechts neben den einzelnen Menü-Punkten (Dienst-Aufrufe) die zu zahlenden Kosten, wenn dieser Dienst angewählt wird. Die angezeigten Kosten werden der Tabelle „im_grund_kosten“ entnommen.

ID_button	ID_kostengruppe	Kosten
Immoment1	1	100
Immoment1	2	1500
Immoment2	1	0
Immoment2	2	0
Immoment100	1	100
Immoment100	2	100

Tabelle 3 – Inhalt der Tabelle im_grund_kosten

Jedem Nutzer ist in der Tabelle „im_user_daten“ eine Kostengruppe (ID_kostengruppe) zugeordnet, derzeit gibt es zwei Kostengruppen, die in der Tabelle „im_kosten_gruppen“ stehen. Die Tabelle „im_grund_kosten“ enthält neben dem Dienst (ID_button) und der Kostengruppe (ID_kostengruppe) die entsprechenden Kosten in Cent, somit kann nach Anmeldung des Nutzers in jedem Menü angezeigt werden, welche Kosten der Nutzer für die einzelnen Dienst-Aufrufe zu zahlen hat. Ist ein Eintrag des Attributes Kosten „0“, so wird rechts neben dem Button kein Kosteneintrag angezeigt, da es sich um ein Menü handelt (in „Tabelle 3“ ist immoment2 das Hauptmenü).

7.1.4 Das Billing-System

Wurde ein Dienst ausgewählt, so werden die entsprechenden Daten angezeigt und es wird gleichzeitig ein Eintrag in den Kostentabellen (Billing-System) vorgenommen. Das Billing-System wird benötigt, um sowohl für die Kunden Rechnungen erstellen als auch mit den Partnern abrechnen zu können.

7.1.4.1 im_kosten_kopf (Kopf-Tabelle)

In der SQL-Tabelle im_kosten_kopf (Kopf-Tabelle) wird direkt nach der Adressprüfung ein Datensatz angelegt, der grundsätzliche Informationen über den Nutzer (Nutzer-Kennung und IP-Adresse) und die Immobilie (hausid) enthält. Diese Daten werden für alle Dienst-Aufrufe einer Adresse nur einmal benötigt und haben den folgenden Aufbau:

ID	ID_user	zeitstempel	waehrung	ip_adresse	hausid	zusatz
678	12	08.04.2002 15:25:00	EUR	127.0.0.1	053140000195400021	

Tabelle 4 – Inhalt der Tabelle im_kosten_kopf

Die Attribute der Kopf-Tabelle enthalten folgende Werte:

- ID Primärkey - laufende Nr (wird automatisch vergeben)
- ID_user Fremdschlüssel – ID in „im_user_daten“
- zeitstempel Tag und minutengenaue Uhrzeit der Anlage
- waehrung Währung (grundsätzlich EUR bzw. €) kommt aus den Nutzer-Daten
- ip_adresse IP-Adresse des angemeldeten Nutzers
- hausid ermittelte hausid durch die Adressprüfung von der abis ag

7.1.4.2 im_kosten_pos_fehler (Fehlersätze)

In der SQL-Tabelle „im_kosten_pos_fehler“ (Fehlersätze der Positionstabelle) wird nur ein Datensatz erstellt, wenn Fehler bewirken, dass kein „im_kosten_pos“ Datensatz angelegt werden konnte oder durfte. Der Aufbau ist identisch mit dem Aufbau der „im_kosten_pos“ Tabelle. Auch die Werte werden synonym angelegt, nur dass in dem Feld „objekt_info“ jeweils ein TAG „<error>“ den Eintrag umrahmt.

7.1.4.3 im_kosten_pos (Positionstabelle)

In der SQL-Tabelle „im_kosten_pos“ (Positionstabelle) wird nach jedem erfolgreichen Dienst-Aufruf ein Datensatz angelegt – sie hat folgenden Aufbau:

ID	ID_Kopf	zeitstempel	ID_button	kosten	Objekt_info
820	678	08.04.2002 15:25:00	immoment1	1500	<plz/><ort>Bonn</ort><ortsteil/> <strasse>Oxfordtr. 21</strasse>
821	678	08.04.2002 15:26:00	immoment500	150	<marktdaten><gkz>05314000 </gkz></ marktdaten>

Tabelle 5 – Inhalt der Tabelle im_kosten_pos

Nach jedem Dienst-Aufruf (auch die Adressprüfung stellt einen Dienst-Aufruf dar) wird ein Datensatz angelegt, dabei wird die „ID“ automatisch vergeben. Das Attribut „ID_Kopf“ enthält die „ID“ des entsprechenden „im_kosten_kopf“ Datensatzes, somit ist ein Bezug (Relation) zu den Grunddaten möglich.

Die Felder der Positionstabelle enthalten folgende Werte:

- ID Primärkey - laufende Nr (wird automatisch vergeben)
- ID_Kopf Fremdschlüssel - ID in „im_kosten_kopf“
- zeitstempel Tag und Uhrzeit der Anlage
- ID_button Fremdschlüssel - ID in „im_button_info“
- kosten den Währungsbetrag (in Cent) aus „im_grund_kosten“
- objekt_info spezifische Daten für die jeweiligen Dienstaufrufe

Um mit den Partnern abrechnen zu können, müssen diesem bestimmte Informationen übergeben werden, z.B. benötigt der Bildlieferant die Angabe, welches Landesvermessungsamt (layer) das Bild geliefert hat. Das machte es nötig, unterschiedliche Werte speichern zu können.

Aus diesem Grund werden die Daten im Feld „objekt_info“ in der Form eines XML-Stromes angelegt. Für die einzelnen Dienst-Aufrufe sind folgende Werte zu speichern:

Adressprüfung (immoment1)

Adress-Eingabe des Nutzers (eingegebene PLZ, Ort und Strasse)

Objektumfeld (immoment100)

- ha_gk HAUS_Koo_Genauigkeit (aus Tabelle: haus_kern)

- ha_qu HAUS_Koo_Quelle (aus Tabelle: haus_kern)

- st_gk STRAB_Koo_Genauigkeit (aus Tabelle: strab_kern)

- st_qu STRAB_Koo_Quelle (aus Tabelle: strab_kern)

Stadtplan (immoment201)

Topo-Karte (immoment206)

Lageplan (immoment202)

Luftbilder (immoment204)

Aufbau ist für alle Bildinformationen gleich

- sy Symbol auf dem Bild (Punkt, Quadrat oder Linie)

- la layer – welches Vermessungsamt in welchem Bundesland

- co comment – Klartext-Hinweis von TMS

- re resolution – Auflösung (Bildqualität)

Objektwert (leih001)

- Kennung, ob OK, da erst später der Aufbau der LEIH-Daten feststeht

Bodenrichtwerte (immoment400)

Mietspiegel (immoment1200)

Aufbau ist für beide gleich -

Datum der Anlage der Daten

Marktdaten (immoment500)

- gkz Gemeinde-Kennziffer (strab_gkz aus Tabelle: strab_basis)

- gkz_ort Ortsbezeichnung

Risiken (immoment600)

- ri HAUS_Risiko (aus Tabelle: haus)

- uv HAUS_Umzugsvolumen (aus Tabelle: haus)

- us HAUS_Umzugssaldo (aus Tabelle: haus)

- fl HAUS_Fluktuatation (aus Tabelle: haus)

Firmendaten (immoment700)

Anzahl der gefunden Firmen in dem Objekt

Branchenumfeld (immoment1000)

Derzeit enthält es nur die Kennung, ob der Dienst angewählt wurde

7.1.5 Kompetenzen steuern die Dienst-Aufrufe

Zur Zeit des Pflichtenheftes wurde die Berechtigung einzelne Dienste aufrufen zu dürfen noch durch einen Eintrag in der Nutzer-Tabelle „im_user_daten“ geregelt (s. Anhang A – 4. Produkt-Funktionalität - Punkt b Objektdaten). Als später die Aufrufe der Web-Services hinzukamen, wurde eine differenziertere Struktur benötigt. Die Kompetenz wird nicht mehr direkt in der Nutzer-Tabelle verwaltet, da ein Nutzer nun mehreren Kompetenz-Gruppen zugeordnet werden kann.

Folgende drei Tabellen werden zur Kompetenzverwaltung benötigt:

- „im_kompetenz_gruppen“ – die Tabelle enthält alle aufrufbaren Dienst-Gruppen:

ID	Bezeichnung	Kuerzel
1	Alle Dienste	gr00
2	Stadtplan, Topo-Karte, Lageplan, Luftbilder	gr01
3	Straßenbilder	gr02
4	Offline-Dienst	gr03
5	Wohnumfeld, Risiken, Firmendaten	gr04
6	Bodenrichtwerte, Mietspiegel, Marktdaten	gr05
7	Infrastruktur, Branchenumfeld	gr06
8	Beleihungswerte (Objektwerte)	gr07
9	Wirtschaftsauskunft	gr08

Tabelle 6 – Inhalt der Tabelle im_kompetenz_gruppen

- „im_kompetenz“ – diese Tabelle führt nun alle Dienste bzw. Programmaufrufe einzeln auf, die in der Tabelle „im_kompetenz_gruppen“ zusammengefasst sind. Der zusammengesetzte Primary-Key besteht dabei aus der „ID“ der Tabelle „im_kompetenz_gruppen“ und der „ID“ der Tabelle „im_button_info“:

ID_kompetenz	ID_button
6	immoment1200
6	immoment400
6	immoment500

Tabelle 7 – Inhalt der Tabelle im_kompetenz

- „im_user_kompetenz“ – diese Tabelle enthält nun die Übersicht, welche Nutzer („ID“ der Tabelle „im_user_daten“), welche Dienste („ID“ der Tabelle „im_kompetenz_gruppen“) aufrufen dürfen.

7.1.6 im_grund_einstellungen

Diese Tabelle enthält eine erweiterbare Liste von Daten, die an vielen Stellen des Programmablaufes benötigt werden. Dazu gehören die IP-Adressen der Partner-Server und Zugangsberechtigungen, die es IMMOMENT erlaubt, von den Partner-Servern Daten herunterzuladen. Ein Eintrag enthält auch die Internet-Seite, die nach dem login aufgerufen wird. Das zeigt, dass die Seitensteuerung von IMMOMENT variabel gehalten wurde. Das macht es leichter, spezielle Seiten anzusteuern.

7.1.7 Administrations-Bereich

Der Administrations-Bereich ermöglicht neben den Anpassungen aller Tabellen, für die Schreibrechte bestehen (durch die SQL-Befehle Update, Insert oder Delete), auch den Ausdruck der Rechnungen an die Kunden.

7.1.7.1 im_user_erste_seite und im_status

Das Erreichen des Administrations-Bereiches wird durch die Tabelle „im_user_erste_seite“ gesteuert. Wie bereits beschrieben wird die Startseite (nach dem Login) der Tabelle „im_grund_einstellungen“ entnommen. Zuerst wird allerdings geprüft, ob für den Nutzer eine spezielle Startseite in der Tabelle „im_user_erste_seite“ vorgegeben wurde. Sollte das der Fall sein, wird der Nutzer auf diese spezielle Seite weitergeleitet. Neben dem Attribut „ID_user“ (die angemeldeten Userkennung) und dem Attribut „ID_button“ (die zu startenden Internet-Seite) enthält die Tabelle noch das Attribut „ID_status“ (Berechtigung). Die verschiedenen Berechtigungen werden in der Tabelle „im_status“ definiert. Derzeit gibt es drei Stati: 0 = Demo-User, 1 = normaler User und 2 = spezieller User. So kann in der Tabelle „im_user_erste_seite“ auch die normale Startseite immoment1 eingetragen sein, aber mit dem Status 0, das hat zur Folge, dass der so berechtigte User keine Adresseingabe vornehmen, sondern nur vorgegebene Adressen aus einer Liste auswählen kann.

7.1.7.2 im_se_admin

Die Tabelle „im_se_admin“ enthält eine Liste der im Administrations-Bereich aufrufbaren Internet-Seiten und welche Tabellen auf den Seiten bearbeitet werden können.

7.1.8 im_hilfe_texte

Die Tabelle "im_hilfe_texte" enthält alle Hilfetexte in Form von HTML-Code, die auf der speziellen Hilfe-Seite „immoment_h“ angezeigt werden. Diese Hilfe-Seite kann von jeder Internet-Seite in IMMOMENT aufgerufen werden. Wird die Hilfe-Seite aufgerufen, so wird ihr in der URL übergeben, welche Internet-Seite sie aufgerufen hat. Daraufhin fordert die Hilfeseite den entsprechenden Hilfetext aus der Tabelle „im_hilfe_texte“ durch einen SQL-Befehl an. Dieses Vorgehen macht es möglich, dass ohne Programm-Änderung ein Hilfetext verändert werden kann. Dadurch dass der Hilfetext als HTML-Code hinterlegt wurde, kann über die Tabelle „im_hilfe_text“ auch eine Schaltfläche (Button) in die Hilfeseite eingesetzt werden, die es ermöglicht, z.B. eine komplette Hilfedatei herunterzuladen, oder die auf andere Seiten verlinkt.

7.1.9 im_kommunikation und im_kommunikations_art

Diese Tabellen ermöglichen die Speicherung von Zusatzdaten für jeden Nutzer, wie Telefon, Fax, Handy oder eMail-Adresse.

7.1.10 Protokoll-Tabellen und Fehlermeldungen

Es zeigte sich, dass es notwendig ist, bestimmte Abläufe zu protokollieren. So wurden mehrere Protokoll-Tabellen angelegt.

7.1.10.1 db_protokoll, click_protokoll und im_zugriffe

In der Tabelle „db_protokoll“ wird festgehalten, welche Tabellen im Administrations-Bereich verändert wurden und von welchem Nutzer.

Die Tabelle „click_protokoll“ speichert die Partner-Server-Aufrufe; insbesondere wenn Fehlermeldungen der Partner-Server zurückkommen, werden diese in der Tabelle „click_protokoll“ festgehalten.

Die Tabelle „im_zugriffe“ hält fest, wie viele Nutzer sich insgesamt angemeldet haben und wie oft Demo-Dateien heruntergeladen wurden.

7.1.10.2 im_error_messages

Diese Tabelle wird speziell für die Web-Services benötigt und enthält Fehlermeldungen, die an den aufrufenden Client zurückgeschickt werden.

7.1.11 im_partner

Die Tabelle „im_partner“ enthält eine Liste aller Partner. Die Eintragungen werden für den Aufbau der Partner-Seite „Abbildung 2“ benötigt.

7.1.12 im_budget und im_budget_user

Die Tabelle „im_budget“ enthält derzeit zwei Budgets, wodurch pro Nutzer in der Tabelle „im_budget_user“ zwei Budgets und die jeweils erreichten Gesamt-Kosten überwacht werden können. Das ermöglicht insbesondere bei Test-User, den Zugang sperren zu können, wenn die Gesamt-Kosten das jeweilige Budget überschreiten.

7.1.13 im_firmen_daten und im_firmen_kostenstellen

Jeder User ist einer Firma zugeordnet, die Firmen werden in der Tabelle „firmen_daten“ verwaltet. Neben dem Namen der Firma enthält die Tabelle „im_firmen_daten“ auch die Anzahl maximaler Nutzer, die dieser Firma zugeordnet werden dürfen (Attribut „anzahl_lizenzen“), das wird bei der Anlage eines neuen Nutzers überwacht. Einige Kunden wünschten, dass die Nutzer einer bestimmten Kostenstelle zugeordnet werden können. Aus diesem Grunde entstand die Tabelle „im_firmen_kostenstellen“, die „ID“ des entsprechenden Datensatzes muss der Nutzertabelle („im_user_daten“) im Feld „ID_kostenstelle“ zugeordnet werden.

7.1.14 im_kunde_seit

Die Nutzer-Verwaltung in IMMOMENT behandelt Test-User und Kunden gleich.

Test-User oder Interessenten bekommen eine verkürzte Nutzungszeit (üblicherweise zwei Wochen), die bei Vertragsabschluss verlängert wird, wodurch ein Interessent zum Kunden wird.

Um zwischen Kunden und Interessenten unterscheiden zu können, wurde die Tabelle „im_kunde_seit“ angelegt, die neben der „ID“ der Firma („ID_firma“) das Datum enthält, seit dem diese Firma Kunde ist.

Durch den Eintrag in dieser Tabelle wird bei Erstellung der Rechnungen erkannt, welche Firmen, seit welchem Datum Kunde sind und nur für diese Firmen werden, ab dem entsprechenden Datum, Rechnungen erstellt.

7.1.15 im_adresse

In diese Tabelle kann für Firmen und einzelne User eine Adresse hinterlegt werden. Die „ID“ der Adresse muss dem Eintrag der Nutzer-Tabelle „im_user_daten“ bzw. „im_firmen_daten“ in dem Attribut „ID_adresse“ zugeordnet werden.

7.1.16 im_druck_berechtigt

Einige Dienste können ausgedruckt werden, das ist bisher aber nur speziellen Kunden vorbehalten. Nur die Nutzer, deren Firmen-Kennung „ID_firma“ in dieser Tabelle eingetragen ist, können z.B. das Wohnumfeld ausdrucken, indem dem Nutzer ein Button „Druck Auswahl“ zur Verfügung steht. Nach Auswahl des Buttons wird eine spezielle Seite angezeigt, die direkt ausgedruckt werden kann.

7.1.17 im_kunden_teleinfo_pic_lura

Ein Partner liefert die Straßenbilder des Dienstes Objektlage, diese können erst seit Anfang 2003 auch im JPEG-Format angezeigt werden. Das JPEG-Format ist für alle Kunden voreingestellt, das hat aber den Nachteil, dass die Bilder etwas größer sind und bei einigen Kunden längere Ladezeiten hervorrufen. Die Bilder können aber auch in einem speziellen Format geladen werden, dazu muss auf dem Client-Rechner aber ein ActiveX installiert sein. Kunden, die dieses spezielle Format wünschen, müssen in dieser Tabelle eingetragen werden und erhalten dann das spezielle Bildformat.

7.2 Die Datenbank im_objekt

IMMOMENT benötigt in der Datenbank „im_objekt“ keine Schreibrechte, denn hier werden ausschließlich die Daten der Partner verwaltet. In regelmäßigen Abständen beliefern die Partner IMMOMENT mit komplett neuen Daten, in einem solchen Fall wird vom Administrator der Datenbank ein Backup eingespielt, wodurch die alten Daten überschrieben werden. Die Web-Applikation greift nur lesend auf die Tabellen der Datenbank „im_objekt“ zu. Exemplarisch wird der Datenbestand eines Partners als Datenbankschema angezeigt (s. „Abbildung 8“). Aus deren Tabellen wird nicht nur der Textblock „Wohnumfeld“ gebildet, sondern auch die Dienste Firmendaten, Infrastruktur, Branchenumfeld und der Textblock Risiken.

7.2.2 im_teleinfo_staedte

Neben dem Wohnumfeld beinhaltet die Datenbank „im_objekt“ weitere Partnerdaten bzw. Übersichtstabellen über die Daten der Partner.

Die Tabelle „im_teleinfo_staedte“ ist eine Übersichtstabelle, sie enthält neben den 46 Städten, in denen die Bilder angefordert werden können, auch zusätzlich das Datum (Jahr) der Aufnahme und wie viele Kilometer Straßen in dem Ort gefilmt worden sind. Nach der Adressprüfung wird in dieser Tabelle nachgelesen, ob für den entsprechenden Ort ein Eintrag vorliegt. Konnte die Stadt in der Tabelle gefunden werden, so sind die beiden Dienste „City-View“ (s. „10.2“) und „City-Server“ (s. „10.3“) anwählbar. Es kann allerdings vorkommen, dass für die entsprechende Straße in der Stadt keine Bilder vorliegen. Der Grund ist darin zu finden, dass der Partner innerhalb einer Stadt noch nicht alle Straßen abfahren konnte. Sollten keine Bilder vorhanden sein, wird das dem Nutzer angezeigt und entsprechend werden keine Kosten abgebucht.

7.3 Die Datenbank im_leih

Die Datenbank „im_leih“ enthält die Daten, die vom Dienst „Objektwert“ angezeigt werden. Die Daten für diesen Dienst stammen aus Beleihungs- oder Verkehrswertermittlungen der Kunden. Sie liefern ihre selbst erfassten Daten an die IMMO-CHECK GmbH, damit diese für IMMOMENT aufbereitet und im Dienst „Objektwert“ angezeigt werden können. Die Idee ist, dass durch die Rücklieferung der einzelnen Kunden ein großer Datenpool aufgebaut werden kann. Jeder Kunde profitiert dadurch von den Daten der anderen Kunden.

Die Kunden, die Daten einliefern, erhalten einen finanziellen Anreiz. Jede Adressprüfung, also der Einstieg um die weiteren Dienste nutzen zu können ist deutlich preiswerter als für die Kunden, die keine Objektwert-Daten einliefern.

Nachdem ein Kunde Beleihungswertdaten eingeliefert hat, müssen die einzelnen Datensätze über die Adresse um die „Haus-ID“ erweitert werden, damit nach der Adressprüfung festgestellt werden kann, ob für die Adresse eine Beleihungs-/Verkehrswertermittlung vorliegt.

Der Nutzungsvertrag von IMMOMENT enthält folgenden Hinweis:

„Teilnehmer des Datenpools können alle Unternehmen sein, die gewerbsmäßig Beleihungs-, Verkehrs- und/oder Objektwerte erstellen (z.B. Banken, Bausparkassen, Versicherungen, die Immobilienwirtschaft, Finanz- und Immobilienmakler). ... Datenrücklieferung im Sinne der ermäßigten Grundinformation [...] bedeutet, der Vertragspartner liefert als Mindestrücklieferung durchschnittlich zu 25% seiner Anfragen auch Daten zu den Objektdaten zurück. Hier wird mindestens die Angabe zu Grundstücksgröße und –preis, Wohn-, Gewerbe- und Nutzflächen mit Objekt-ausstattung und Verkaufspreis erwartet. ... Insbesondere bei der Datenrücklieferung dürfen ausschließlich berechnete Personen tätig werden. Bei dem Zugriff auf spezielle Daten, wie z.B. Beleihungswerte, ist ein berechtigtes Interesse nachzuweisen.“

Die „Abbildung 9“ stellt die Startseite (Objektart) der Verkehrswert-/Beleihungswertdaten (im Demo-Modus) dar, die durch acht weitere Web-Seiten ergänzt wird.

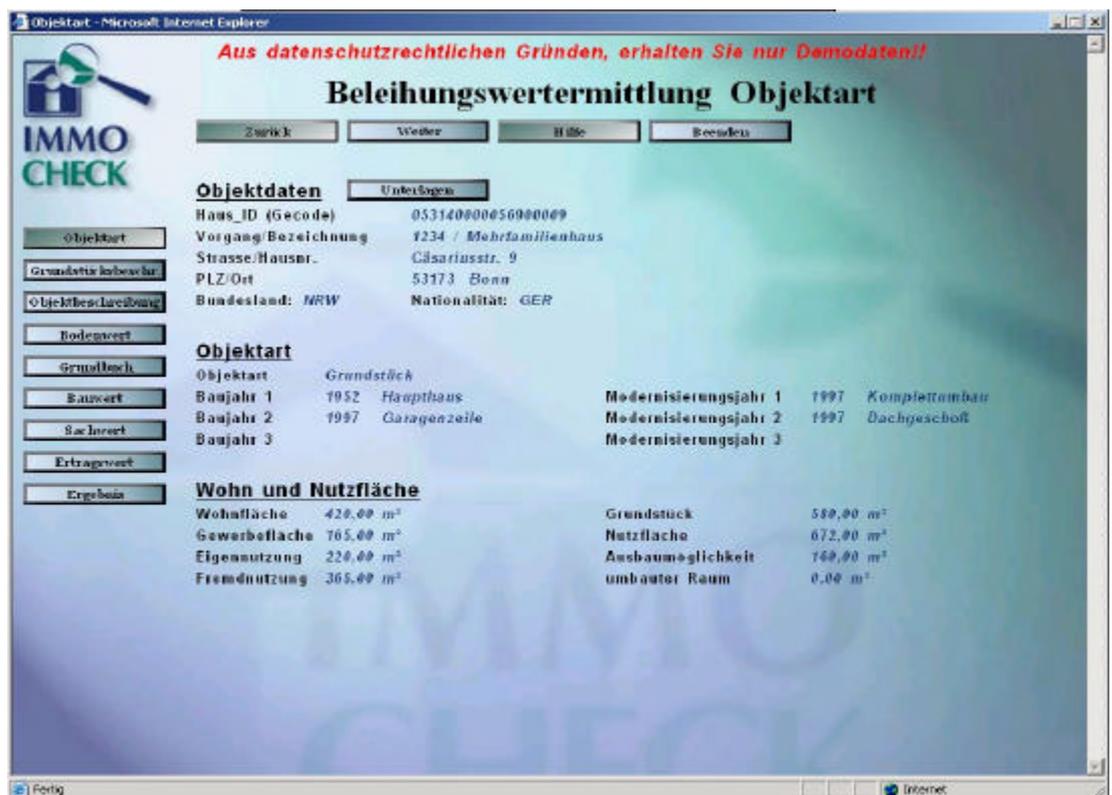


Abbildung 9 – Web-Ansicht des Dienstes „Objektwert“

Der angesprochene Datenpool hat derzeit erst einige Tausend Datensätze, aus diesem Grunde liegen für eine eingegebene Adresse sehr selten Objektwerte vor. Daraus ergibt sich, dass der Dienst, wie in „Abbildung 1“ zu erkennen, selten anwählbar ist.

7.4 Die tempdb verwaltet die Session

Im ersten Kapitel unter „1.4 Arten von Internetseiten“ wurde bereits vorgestellt, dass u.a. durch serverseitig verwaltete, dynamische Internetseiten Informationen einer Browser-Anfrage gespeichert werden können. Die in IMMOMENT verwendete Technologie ASP.Net ermöglicht, dass der Server pro Client einen bzw. mehrere Speicherbereiche verwaltet. Dieser Speicherbereich wird auch Sitzung oder englisch „Session“ genannt.

Pro gestartetem Browser eines Client-Rechners wird eine Session gespeichert, damit diese Session vom Server verwaltet wird, muss im C#-Programm nur ein so genanntes Session-Objekt durch den write-Befehl angelegt werden. In IMMOMENT werden die Daten, die in der Session verwaltet werden, als XML-Strom gespeichert, dadurch kann semistrukturiert schnell wieder auf die Daten zugegriffen werden. ASP.Net stellt für XML-Ströme bereits Methoden zur Verfügung, wodurch die Daten einfach ausgelesen werden können.

Die Session-Verwaltung kann aber nicht komplett serverseitig geschehen, denn bei der nächsten Anfrage des Browser muss der Server genau die richtige Session zur Verfügung stellen. Um dies zu gewährleisten ist ein Schlüssel nötig, der vom Client-Browser an den Server übergeben wird. Durch diesen Schlüssel wird genau die richtige Session ermittelt, dieser Schlüssel nennt sich Session-ID¹¹⁷.

¹¹⁷ Die **Session-ID** stellt einen READ_ONLY-Wert dar. Mit der Session-ID wird ein Nutzer gegenüber dem Web-Server eindeutig identifiziert. Bei ASP.Net besteht die Session-ID aus einer Folge von 120-Bit ASCII-Zeichen. Die Dokumentation von MS gibt an, dass die Session-IDs mittels eines Algorithmus erzeugt werden, der ihre Einzigartigkeit garantiert. Zusätzlich wird ein Zufallsfaktor verwendet, der es erheblich schwieriger macht, die ID einer bestehenden Session bestimmen zu können, wodurch die Sicherheit erhöht wird. Die Session-ID wird in ASP.Net normalerweise als temporärer Cookie auf dem Client-Rechner abgelegt. Neben der traditionellen Methode des Stuserhalts durch temporäre Cookies wird von ASP.Net eine „cookiefreie“ Variante unterstützt. Wurde der cookiefreie Modus gewählt, verwaltet ASP.Net die Session-ID in die URL, die an den Client zurückgeschickt wird. Dadurch arbeiten auch Browser zuverlässig bei denen die Cookies nicht aktiviert sind.

Daraus ergeben sich zwei zu berücksichtigende Fragestellungen:

1. Wo soll die Session serverseitig verwaltet werden? Grundsätzlich bestehen zwei Alternativen die Session serverseitig zu verwalten: Erstens kann sie direkt vom Web-Server selbst verwaltet werden oder zweitens von einem speziellen Datenbank-Server. Die Web-Applikation IMMOMENT wird von einer Web-Farm, also mehreren Web-Servern gesteuert wird, wobei durch ein Hardware „load balancing“ die Last automatisch verteilt wird. Das Vorgehen steigert die Verfügbarkeit, es muss aber gewährleistet werden, dass eine neue Anfrage auch an die bereits bestehende Session gelangt, die aber von einem anderen Server angelegt worden sein kann. Aus diesem Grunde musste für IMMOMENT die Session von einem speziellen Datenbank-Server verwaltet werden. Dieser DB-Server muss jedem Web-Server bekannt sein, damit die Session von dem DB-Server gelesen und geschrieben werden kann.
2. Wie sollte die Session-ID clientseitig gespeichert werden? Auch für diese Frage gibt es zwei grundsätzliche Antworten: Erstens kann die Session-ID in einem Cookie auf dem Client-Rechner gespeichert werden, oder zweitens die Session-ID wird in die URL eingebettet, eine Browser-Anfrage kann dadurch folgendermaßen aussehen: [www.immoment.com/\(mhcjg0il0dh2efaueube3g45\)/immoagent.aspx](http://www.immoment.com/(mhcjg0il0dh2efaueube3g45)/immoagent.aspx) Der Teil in den runden Klammern „(...)“ ist die Session-ID, die nun bei jeder Anfrage des Browsers an den Server übergeben wird, ASP.Net ermöglicht dann serverseitig das Auslesen der Session-ID. Grundsätzlich bestand für IMMOMENT keine spezielle Vorgabe, da es aber von Vorteil ist keine Cookies auf dem Client-Rechner einzusetzen, wurde die zweite Variante ohne Cookies gewählt.

Interessanterweise müssen beide Fragestellungen nicht unbedingt vor der Implementierung der Web-Anwendung bedacht werden, denn ASP.Net arbeitet mit zwei XML basierenden, hierarchisch aufgebauten Konfigurationsdateien, die machine.config und die web.config. Die machine.config stellt die Hauptkonfigurationsdatei dar, von der sich alle anderen Konfigurationsdateien ableiten. Aus diesem Grunde existiert die machine.config auch nur einmal auf jedem Rechner und enthält die standardmäßigen Einstellungen für den jeweiligen Computer. Die web.config hingegen besteht für jede Web-Applikation mindestens einmal, es kann aber auch für jede Unterseite eine weitere angelegt werden. Beim Start der Applikation bzw. der Unterseiten wird zuerst die entsprechende web.config interpretiert, das ermöglicht Anpassungen während der Laufzeit. Beide Vorgaben (für den Ort der Session und der Session-ID) verwaltet die web.config. Beispielhaft wird die web.config von IMMOMENT aufgeführt:

```

<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <system.web>
    <compilation defaultLanguage="c#" debug="true"/>
    <customErrors mode="Off"/>
    <authentication mode="None"/>
    <authorization> <allow users="*" /> </authorization>
    <trace enabled="false" requestLimit="10"
      pageOutput="false" traceMode="SortByTime" localOnly="true"/>
    <sessionState118
      mode="SQLServer" stateConnectionString="tcpip=127.0.0.1:42424"
      sqlConnectionString="data source=drzasga;user id=writer;password=x2"
      cookieless="true" timeout="20"/>
    <globalization requestEncoding="utf-8" responseEncoding="utf-8"/>
  </system.web>
  <appSettings119>
    <!-- 1. SQL-SERVER Vorgaben – Datenbankzugriffe ..... -->
    <add key="im_basis"
      value="server=drzasga;database=im_basis;UID=writer;pwd=x2"/>
    <add key="im_objekt"
      value="server=drzasga;database=im_objekt;UID=reader;pwd=y2"/>
    <add key="im_leih"
      value="server=drzasga;database=im_leih;UID=reader;pwd=y2"/>
    <!-- 2 Systemumgebung ..... -->
    <add key="web_proxy" value="" />
    <add key="error_page" value="error.aspx"/>
    <add key="IMMOMENT.tms_getdata.XDOTNET"
      value="http://195.27.54.40/xdotnet/xdotnet.asmx"/>
    <add key="debug_path" value="c:\inetpub\debug\" />
  </appSettings>
</configuration>

```

¹¹⁸ <SessionState> Konfiguriert den Session State, hier ist angegeben, dass die Session im Datenbank-Server (mode="SQLServer") und die Session-ID ohne Cookies (cookieless="true") verwaltet wird.

¹¹⁹ <appSettings> Hier können benutzerdefinierte Vorgaben wie z.B. Datenbankzugriffe eingetragen werden. Erweitert wird die Liste durch ein zusätzliches TAG <add key="" value="">. Dieser Standard-Konfigurations-Handler ist sowohl in der machine.config als auch in allen web.configs zulässig.

Sollte beschlossen werden, dass ein spezieller Datenbank-Server die Session verwaltet, und die entsprechenden Eintragungen wurden in der web.config vorgenommen, so ist zu bedenken, dass noch ein spezielles Konfigurations-Skript im SQL-Server ausgeführt werden muss. Dieses Konfigurations-Skript wird durch die Installation vom .Net-Framework mitgeliefert und auf die Festplatte kopiert, sie heißt „InstallSqlState.sql“ und legt nach Ausführung eine Datenbank an und erweitert die temporäre DB:

1. Die Datenbank ASPState wird angelegt, sie hat keine eigenen Tabellen, aber mehrere „stored Procedures¹²⁰“, die zur Session-Verwaltung benötigt werden.
2. In der bereits vorhandenen Datenbank tempdb werden zwei Tabellen angelegt (ASPStateTempApplikations und ASPStateTempSessions), in der pro Applikation die Session der einzelnen User verwaltet werden.

Aufbau der ASPStateTempApplikations (speichert einen Eintrag pro gestarteter Applikation):

<u>Feldname</u>	<u>Datentyp</u>
AppId	int
AppName	char(28)

Aufbau der ASPStateTempSessions (speichert einen Eintrag pro angelegter Session):

<u>Feldname</u>	<u>Datentyp</u>
SessionId	int
Created	datetime
Expires	datetime
LockDate	datetime
LockCookie	int
Timeout	int
Locked	bit
SessionItemShort	varbinary(7000)
SessionItemLong	image(16)

Der in der web.config unter <sessionState> vorgegebene Teil „timeout“ legt fest, wie lange eine Session aktiv bleibt, solange keine Anfrage vom Client-Browser ausgelöst wird. Dieser Eintrag wird in der ASPStateTempSession als „Timeout“ übernommen.

¹²⁰ **Stored Procedures** (gespeicherte Prozeduren) sind Datenbank-Objekte, die unabhängig von Tabellen sind und Variablen enthalten können. Sie werden vorkompiliert in der DB gespeichert.

Damit nun die Session automatisch gelöscht wird, wenn der entsprechende Timeout-Eintrag (z.B. 20, das steht für 20 min) erreicht wurde, muss über den Enterprise-Manger des SQL-Servers noch der „SQL-Server-Agent“ aktiviert werden. Dieser Manager befindet sich im Bereich „Verwaltung“. Nachdem er aktiviert wurde, ein grüner Pfeil kennzeichnet dies, überwacht der SQL-Server die Session-Einträge und löscht die, deren Timeout-Eintrag erreicht wurde.

Welche zusätzlichen Informationen (s. <appSettings>) in der web.config verwaltet werden, muss aus Sicht des Providers betrachtet werden, auf dessen Web-Servern die Applikation eingesetzt wird. Somit ist es sicherlich sinnvoll, die Datenbankzugriffe in der Konfigurationsdatei zu verwalten. Sollte der Provider genötigt sein einen anderen Datenbank-Server einzusetzen oder die Zugriffsrechte zu ändern, so ist es von Vorteil, wenn er diese Änderungen auch entsprechend in der web.config anpassen kann. Hingegen sollten Zugriffsrechte oder IP-Adressen anderer Server, die von der Web-Applikation aufgerufen werden, nicht in der web.config verwaltet werden, denn dann muss bei einer Änderung zuerst der Provider informiert werden und erst nach Änderung der web.config arbeitet die Web-Anwendung wieder korrekt.

Es lässt sich leicht erkennen, dass in der web.config auch sicherheitssensitive Daten als Klartext angeführt sind. Aus diesem Grund konfiguriert ASP.Net den IIS, mit dem Ergebnis, dass ein Browser auf Dateien mit der Endung „.config“ nicht direkt zugreifen kann und somit deren Information nicht in falsche Hände kommt.

8 Wohnumfelddaten

Im zweiten Kapitel „Techniken“ wird u.a. erläutert, wie „2.1 Die Skript-Sprache“ von IMMOMENT funktioniert und warum diese entwickelt wurde.

Das zu interpretierende Skript für das Wohnumfeld (s. Abbildung 8) steht dabei in einer Steuerungstabelle. Der Aufbau des Wohnumfeld-Textes wird durch eine solche Steuerungstabelle geregelt – der Tabelle „zeilen_steuerung_wup“.

Dieses Kapitel erläutert den Aufbau der Steuerungstabelle und die dazu benötigten weiteren Tabellen und veranschaulicht somit die Struktur der Wohnumfeld-Texte.

8.1 Eine gekürzte Beispiel-Ausgabe

Das Objekt ist ein: Überwiegend gewerblich genutztes Haus

Für dieses Objekt sind bekannt: 3 Privatadressen und 8 Gewerbeadressen.

Detailliertere Informationen zu den im Objekt ansässigen Firmen bietet das Modul "Firmendaten".

Die Beschreibung des Objektumfeldes basiert hier auf den Daten des Straßenabschnittes

von Hausnummer: 21 bis Hausnummer: 9999 Zählweise: nur ungerade Hausnummernseite

Insgesamt liegen in diesem Straßenabschnitt 3 Objekte.

Differenziert nach der Größe des Objektes, die allein durch die Zahl der Haushalte (HH) pro Objekt (ohne Berücksichtigung der darin ansässigen Firmen) ausgedrückt wird, verteilen sich diese Objekte wie folgt:

überwiegend wohnwirtschaftlich genutzte Objekte mit 1-2 HH: 0

überwiegend oder rein gewerblich genutzte Häuser: 2

Objekte unbekannter Größe: 1

Der Straßenabschnitt ist eine: Gewerbestraße

8.2 Steuerungstabelle zeilen_steuerung_wup

Folgender Auszug veranschaulicht den Inhalt der Steuerungs-Tabelle:

ID	zeile
1001	Das Objekt ist ein: ##/h2_tabelle[ID='##/haus/HAUS_Haustyp#§']/text#§
1011	##?IF ##/haus/HAUS_Haushalte#§ > 0 or ##/haus_kern/HAUS_Gewerbeinfo#§ > 0#§
1015	Für dieses Objekt sind bekannt: ##?EXCELIF ##/haus/HAUS_Haushalte#§>0;##/haus/HAUS_Haushalte#§ Privatadress#?EXCELIF ##/haus/HAUS_Haushalte#§>1;en;e#§;#§#?EXCELIF ##/haus/HAUS_Haushalte#§>0 and ##/haus_kern/HAUS_Gewerbeinfo#§>0; und ;#§#?EXCELIF ##/haus_kern/HAUS_Gewerbeinfo#§>0;##/haus_kern/HAUS_Gewerbeinfo#§ Gewerbeadress#?EXCELIF ##/haus_kern/HAUS_Gewerbeinfo#§ > 1;en;e#§;#§.
1019	##?ENDIF#§
1020	##?IF ##/haus_kern/HAUS_Gewerbeinfo#§ > 0#§
1021	Detailliertere Informationen zu den im Objekt ansässigen Firmen bietet das Modul "Firmendaten".
1025	##?ENDIF#§
1026	##?REM Mosaic-Wohnumfeldmerkmale #§
1035	##?IF##/STRAB_Basis/STRAB_NummerVon#§ = 0 AND ##/STRAB_Basis/STRAB_NummerBis#§ = 9999#§
1041	Die Beschreibung des Objektfeldes basiert hier auf den Daten der kompletten Straße, auch wenn diese nachfolgend als "Straßenabschnitt" bezeichnet wird.
1043	##?ELSE#§
1046	Die Beschreibung des Objektfeldes basiert hier auf den Daten des Straßenabschnittes
1051	von Hausnummer: ##/STRAB_Basis/STRAB_NummerVon#§ bis Hausnummer: ##/STRAB_Basis/STRAB_NummerBis#§ ##/zaehlweise_texte[ID='##/STRAB_Basis/STRAB_Seite#§']/text#§
1056	##?ENDIF#§
1061	
1066	Insgesamt liegen in diesem Straßenabschnitt ##/STRAB_Basis/STRAB_Haeuser#§ Objekte.
1071	Differenziert nach der Größe des Objektes, die allein durch die Zahl der Haushalte (HH) pro Objekt (ohne Berücksichtigung der darin ansässigen Firmen) ausgedrückt wird, verteilen sich diese Objekte wie folgt:
1076	##?TAB3Überwiegend wohnwirtschaftlich genutzte Objekte mit 1-2 HH:#§ ##?MATH ##/STRAB_Haustyp/STRAB_Haustyp_1#§ + ##/STRAB_Haustyp/STRAB_Haustyp_2#§ #§

Tabelle 8 – Inhalt der Steuerungstabelle zeilen_steuerung_wup

8.3 Die weiteren Tabellen

Alle weiteren Tabellen (s. Abbildung 8) sind von der Firma microm Marketing und Consult GmbH vorgegeben. Die Tabellen liefern die anzuzeigenden Daten. Sind alle Werte für die entsprechende Adressvorgabe eingelesen, kann die Steuerungs-Tabelle zeilen_steuerung_wup die benötigten Werte nachlesen. Zur Veranschaulichung folgen Auszüge aus den angesprochenen Tabellen:

ID	text
D	zählweise: durchlaufend
G	zählweise: nur gerade Hausnummernseite

Tabelle 9 – Inhalt der Tabelle zaehlweise_texte

ID	text
0	keine Angaben
1	reine Wohnstraße

Tabelle 10 – Inhalt der Tabelle s10_tabelle

GKZ_ID	GKZ_ Gemeindenname	GKZ_ Kreisname	GKZ_ Regierungs bezirk	GKZ_ Bundesland	GKZ_ Ortsgrößen klasse	GKZ_ Gemeinde typ
01001000	Flensburg, Stadt	Flensburg, Stadt	<NULL>	Schleswig- Holstein	3	LLZ
01002000	Kiel, Landeshauptstadt	Kiel, Landeshauptstadt	<NULL>	Schleswig- Holstein	2	VKK

Tabelle 11 – Inhalt der Tabelle Gemeinde

ID	Kuerzel	text
1	AHU	Umlandgemeinde in Ballungsregion
2	AHZ	Zentrum in Ballungsregion

Tabelle 12 – Inhalt der Tabelle gemeinde_typ

ID	text
1	500.000 und mehr Einwohner
2	100.000 bis unter 500.000 Einwohner

Tabelle 13 – Inhalt der Tabelle gemeinde_groesse

STRAB_ID	STRAB_dom_Auslaenderklasse	STRAB_dom_Strassentyp	STRAB_Kaufkraft_klasse
0000011000001	3	3	0
0000021000001	5	3	0

Tabelle 14 – Inhalt der Tabelle STRAB_dom_Klassen

STRAB_ID	STRAB_GKZ	STRAB_Seite	STRAB_Numm erVon	STRAB_Zusatz Von	STRAB_Numm erBis	STRAB_Zusatz Bis	STRAB_Haeuser	STRAB_Haushalte
0100100 000010	01001000	D	0	<NULL>	9999	<NULL>	52	55
0100200 000010	01002000	D	0	<NULL>	9999	<NULL>	15	26

Tabelle 15 – Inhalt der Tabelle STRAB_BASIS

STRAB_ID	STRAB_Anteil _Risiko _niedrig	STRAB_Anteil _Risiko _mittel	STRAB_Anteil _Risiko _hoch	STRAB_Anteil _Status _niedrig	STRAB_Anteil _Status mittel	STRAB_Anteil _Status _hoch	STRAB_Anteil Ausland _niedrig	STRAB_Anteil Ausland _mittel	STRAB_Anteil Ausland _hoch
0100100 000010	68	32	0	20	52	28	64	36	0
0100200 000010	17,4	39,13	43,48	21,74	78,27	0	78,26	21,74	0

Tabelle 16 – Inhalt der Tabelle STRAB_RISIKO_STATUS_AUSLAND

STRAB_ID	STRAB_Haustyp_ 1	STRAB_Haustyp_ 2	STRAB_Haustyp_ 3	STRAB_Haustyp_ 4	STRAB_Haustyp_ 5	STRAB_Haustyp_ 6	STRAB_Haustyp_ 7	STRAB_Haustyp_ G
0100100 000010	0	30	8	1	0	0	0	13
0100200 000010	0	9	4	0	0	0	1	1

Tabelle 17 – Inhalt der Tabelle STRAB_HAUSTYP

STRAB_ID	STRAB_Mosaictyp_ Top1	STRAB_P_Mosaictyp_ Top1	STRAB_Mosaictyp_ Top2	STRAB_P_Mosaictyp_ Top2	STRAB_Mosaictyp_ Top3	STRAB_P_Mosaictyp_ Top3
0100100000010	07	49,09	28	12,73	03	10,91
0100200000010	26	30,77	37	26,92	33	23,08

Tabelle 18 – Inhalt der Tabelle STRAB_MOSAICTYP_TOP3

ID	text
0	Keine Angabe
1	Niedrigste Kaufkraft

Tabelle 19 – Inhalt der Tabelle kaufkraft_texte

ID	text
A	Statushohe Großstädter
B	Gutsituierte in stadtnahen Umlandgemeinden

Tabelle 20 – Inhalt der Tabelle Mosaic_gruppe

ID	gruppe	Typ	text
00	A	Keine Angaben	Keine Angaben
01	A	Attraktive innenstädtische Wohnlage	Luxuriöse Mehrfamilienhäuser in ...
02	A	Wohlhabende Akademiker in Villenvierteln	Höchste berufliche Qualifikation, ...

Tabelle 21 – Inhalt der Tabelle mosaic_typ

KREIS_ID	KREIS_Mosaictyp	KREIS_Mosaictyp_Anteil
01001	01	0,35
01001	02	2,31
01002	01	3,45

Tabelle 22 – Inhalt der Tabelle KREIS

ID	text
1	1-2 Familienhais in homogen bebautem Straßenabschnitt
2	1-2 Familienhais in nicht homogen bebautem Straßenabschnitt
3	3-5 Familienhaus

Tabelle 23 – Inhalt der Tabelle h2_tabelle

ID	text
1	niedrigster Status
2	weit unterdurchschnittlicher Status
3	unterdurchschnittlicher Status

Tabelle 24 – Inhalt der Tabelle h6_tabelle

HAUS_ID	HAUS_Koo_Rechts	HAUS_Koo_Hoch	HAUS_Koo_Genauigkeit	HAUS_Koo_Quelle	HAUS_Wohnumfeld	HAUS_Gewerbeinfo
010010000 010100005	3609496,29	5271126,49	P01	1	1	2
010010000 010100005	3609496,29	5271126,49	P01	1	0	0

Tabelle 25 – Inhalt der Tabelle HAUS_KERN

HAUS_ID	HAUS_Haustyp	HAUS_Haus halte	HAUS_Mosaic typ	HAUS_Status	HAUS_Risiko	HAUS_Umzugs volumen	HAUS_Umzugs saldo	HAUS_Fluktuation
010010000 010100005	2	1	07	1	1	5	5	5
010010000 010100005	G	0	39	G	G	G	G	G

Tabelle 26 – Inhalt der Tabelle HAUS

9 Branchen im Umfeld einer Immobilie

IMMOMENT ist kein Geoinformationssystem (GIS), aber es ermöglicht die Darstellung von Daten, die sich auf Immobilien beziehen. Die Lage-Koordinaten erlauben nicht nur die Anzeige von Kartenausschnitten, Luftbildern und Straßenansichten (s. „10 Der Dienst Objektlage“), sondern sie gestatten auch die Beantwortung der Frage nach dem „Aussehen“ des Umfeld. In diesem Kapitel werden die Dienste beschrieben, die anzeigen, welche Branchen in einem bestimmten Umkreis um die Immobilie liegen, bzw. wie gut ist die Infrastruktur in dem Umfeld ausgebaut ist.

9.1 Der Dienst Infrastruktur

Nach Anwahl des Dienstes „Infrastruktur“ aus dem Hauptmenü dauert es einige Sekunden, bis folgende Ansicht erscheint:

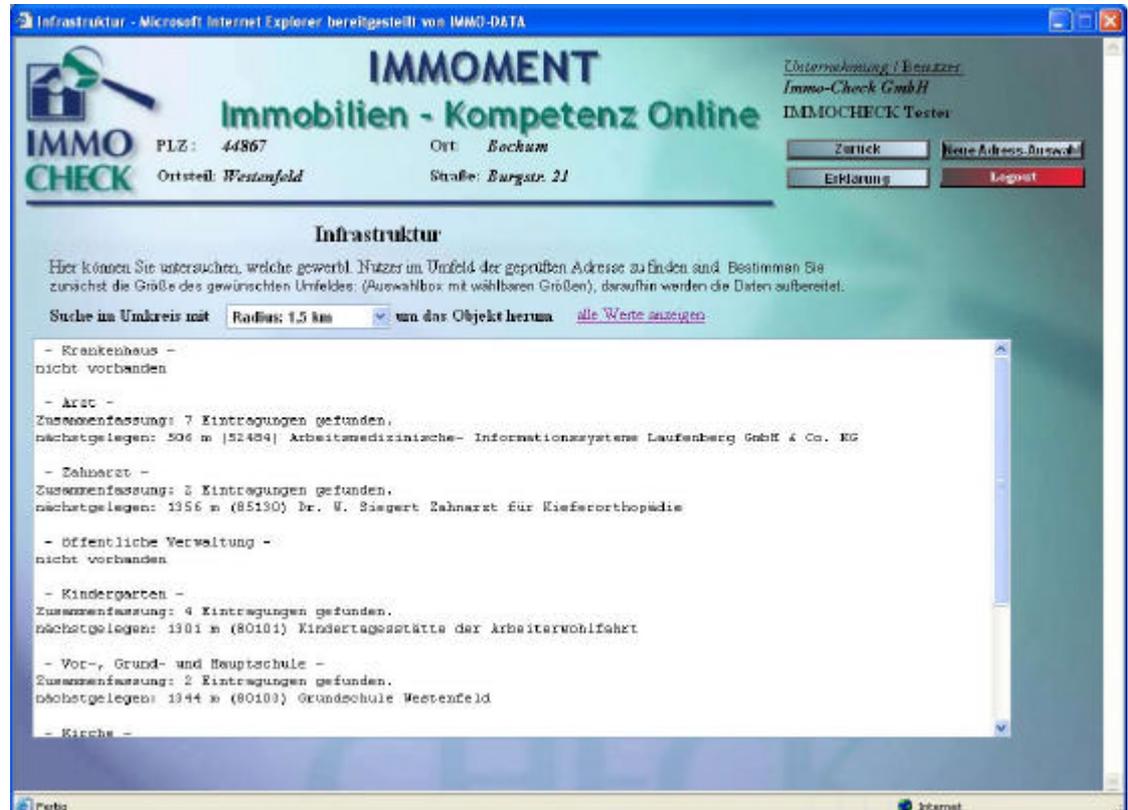


Abbildung 10 – Web-Ansicht des Dienstes Infrastruktur

Folgende Tabellen ermöglichen die Auswahl:

Im_infrastruktur_entfernung: Füllt die Auswahlbox für den Umkreis:

<u>value</u>	<u>text</u>
500	Radius: 500 m
1000	Radius: 1 km
1500	Radius: 1,5 km
2000	Radius: 2 km
5000	Radius: 5 km
10000	Radius: 10 km

Im_infrastruktur_gruppen:

Gibt die Branchengruppen vor:

<u>ID</u>	<u>Bezeichnung</u>
10	Krankenhaus
20	Arzt
21	Zahnarzt
30	öffentliche Verwaltung
40	Kindergarten
usw.	

Im_infrastruktur_branchen:

Gibt die Branchen jeder Branchengruppe vor:

<u>ID</u>	<u>gruppe</u>	<u>branmr (WZ93-Schlüssel)</u>
10		85110
10		85111
10		85112
20		85120
21		85130
usw.		

Zusätzlich werden folgende Tabellen benötigt:

HAUS_KERN

Enthält alle Immobilien-Objekte mit deren Lage-Koordinaten („HAUS_Koo_Hoch“ und „HAUS_Koo_Rechts“), um bestimmen zu können, ob ein Objekt im vorgegebenen Radius liegt. Das Attribut „HAUS_Gewerbeinfo“ zeigt an, ob in diesem Immobilien-Objekt gewerbliche Nutzer vorhanden sind.

FIRMEN

Enthält alle Firmen mit deren Branchengruppen („branmr1“ und „branmr2“)

Die Tabellen „im_infrastruktur_branchen“ und „im_infrastruktur_gruppen“ sind dynamisch erweiterbar.

Ablauf der Datensuche:

Nach Anwahl des Dienstes „Infrastruktur“ werden für den kleinsten Radius (500 m) die vorgegebenen Branchen gesucht („im_infrastruktur_entfernung“ Attribut „value“ - start mit dem 1. Eintrag). Wie die Suche der Daten durchgeführt wird, ist ausführlich im Kapitel Technik „2.4 Datensuche durch Lage-Koordinaten“ beschrieben.

Wurden Einträge gefunden, so wird zuerst die Branchengruppe ausgegeben („im_infrastruktur_gruppen“), die Anzahl der gefundenen Einträge zu dieser Branchengruppe und der nächstgelegene Eintrag wird mit Entfernung zum Ausgangsobjekt, deren ersten Brancheneintrag als WZ93¹²¹-Schlüssel und den Firmennamen angezeigt.

Konnten Einträge gefunden werden, aber nicht für alle vorgegebenen Branchengruppen, so erhalten die Gruppen, zu denen kein Eintrag gefunden wurde, den Hinweis: „nicht vorhanden“.

In diesem Fall kann nun aus der Entfernungsbox ein größerer Radius ausgewählt werden und dadurch wird gleichzeitig eine neue Suche ausgelöst.

Wurden für alle vorgegebenen Branchengruppen Einträge gefunden, so ist es nicht mehr möglich, aus der Entfernungsbox einen anderen Wert auszuwählen.

Sobald Werte gefunden wurden, kann durch die Schaltfläche „alle Werte anzeigen“ nicht nur die Zusammenfassung, sondern jeder gefundene Firmen-Eintrag angezeigt werden.

Antwortzeit des Servers: Die Suche für die einzelnen Werte kann einige Zeit in Anspruch nehmen. Wird für den Radius zehn Kilometer eine Suche gestartet, kann es etwa eine halbe Minute dauern, bis die Werte angezeigt werden, zusätzlich wäre die Liste aufgrund ihrer Länge unübersichtlich. Aus diesem Grunde wurden die Radien fünf und zehn Kilometer für Städte über 100.000 Einwohner gesperrt.

¹²¹ Der fünfstellige Branchenstruktur-Schlüssel stellt die offizielle Systematik der Wirtschaftszweige des Statistischen Bundesamtes (Wiesbaden) dar, der 1993 auf EU-Basis vereinheitlicht wurde und seither kurz **WZ93** genannt wird.

EU (Europäische Union) ist ein wirtschaftlicher und politischer Zusammenschluss von 25 Staaten (Stand: 1.5.2003). Ihre Mitglieder haben der EU bestimmte Souveränitätsrechte übertragen. Deshalb handeln diese Staaten in einigen Bereichen gemeinschaftlich u. fassen Beschlüsse, die für alle verbindlich sind. „Brüsseler“ Entscheidungen bestimmen viele nationale Entscheidungen.

9.2 Der Dienst Branchenumfeld

Dieser Dienst ist ähnlich dem gerade beschriebenen Dienst „Infrastruktur“ aufgebaut. Wie der „Abbildung 11“ entnommen werden kann, zeigt das Branchenumfeld allerdings alle Branchen in einem vorgegebenen Umkreis der Immobilie an. Dabei wird die Anzahl der gefundenen gewerblichen Nutzer (Firmen) ausgegeben und alle entdeckten Branchen in dem gewählten Umkreis werden angezeigt und als Liste aufgeführt. Der erste Eintrag der Branchen-Liste wird automatisch ausgewählt und die erste Firma mit den Kenngrößen Name, Branche, Branchengruppe, Mitarbeiterzahl, Umsatz und (falls vorhanden) der Crefo-Nr aufgeführt.

Sollten zu der angezeigten Branche mehrere Firmen vertreten sein, so wird neben der Anzahl der entdeckten Firmen auch eine weitere Listbox angezeigt, die die entsprechenden Firmen enthält, damit der Nutzer jede einzelne anwählen kann.

Neben der Entfernung (Punktabstandsformel) lässt sich auch die Himmelsrichtung zur untersuchten Immobilie ermitteln, diese Angaben werden ebenfalls ausgegeben.

Branchenumfeld - Microsoft Internet Explorer

IMMOMENT
Immobilien - Kompetenz Online

Unternehmung / Benutzer:
Immo-Check GmbH
IMMOCHECK Tester

PLZ: 53173 Ort: Bonn
Ortsteil: Rüngsdorf Straße: Cäsariusstr. 9

Zurück Home/Adress-Auswahl
Erklärung Logout

Branchen - Umfeld

Hier können Sie untersuchen, welche gewerbliche Nutzer im Umfeld der geprüften Adresse zu finden sind. Bestimmen Sie zunächst die Größe des gewünschten Umfeldes: (Auswahlbox mit wählbaren Größen), daraufhin werden die Daten aufbereitet.

Suche im Umkreis mit Radius: 500 m um das Objekt herum

In dem vorgegebenen Raum gefundene gewerbliche Nutzer: 217

Auswahl nach entdeckten WZ93 - Branchen: 103

52310 - Apotheken

3 gewerbliche Nutzer im Anschnitt zu der Branche wurden gefunden:

Linden-Apotheke Norbert Paul e. K.

Firmenname: Linden-Apotheke Norbert Paul e. K.
Branche: Apotheken
Branchengruppe: Einzelhandel (ohne Handel mit Kraftfahrzeugen und ohne Tankstellen); Reparatur von Gebrauchsgütern
Mitarbeiterzahl: unbekannt
Umsatz: unbekannt
Crefo Nr.: Entfernung und Richtung zum Objekt: 448 m WWesten

Abbildung 11 – Web-Ansicht des Dienstes Branchenumfeld

Folgende Tabellen ermöglichen die Auswahl:

Im_branchenumfeld_entfernung: Füllt die Auswahlbox für den Umkreis:

value	text
50	Radius: 50 m
100	Radius: 100 m
200	Radius: 200 m
500	Radius: 500 m
1000	Radius: 1 km
2000	Radius: 2 km
5000	Radius: 5 km

Wie bei der Infrastruktur werden neben der gerade aufgeführten Entfernungstabelle die Tabellen „HAUS_Kern“ und „firmen“ benötigt. Zusätzlich werden noch die Tabellen „firmen_branchen“ (enthält zu der WZ93 die entsprechende Bezeichnung der Branche), „firmen_umsatz“ (enthält den ungefähren Umsatz in €) und „firmen_mitarbeiter“ (enthält die ungefähre Mitarbeiteranzahl) benötigt.

Die Datensuche geschieht ähnlich der Infrastruktur, allerdings entfällt der Vergleich mit der Branchenliste (Im_infrastruktur_gruppen, bzw. Im_infrastruktur_branchen). Das technische Vorgehen ist ebenfalls ausführlich im Kapitel Technik „2.4 Datensuche durch Lage-Koordinaten“ beschrieben.

10 Der Dienst Objektlage

Es können über den Dienst „Objektlage“ zwei Arten von Bildinformationen angefordert werden. Auf der einen Seite werden Informationen direkt zur Immobilie angezeigt: Stadtpläne, topographische Karte, Lageplan und Luftbilder. Diese Daten stammen von einem Partner und werden durch Aufrufe von Web-Services eingelesen.

Auf der anderen Seite können derzeit aus 46 Städten Bilder angefordert werden (s. 7.2.2 im_teleinfo_staedte), die aus einem fahrenden Fahrzeug mit sechs bis acht Kameras in zehn Meter Abstand (die Straße entlang am Objekt vorbei fahrend) aufgenommen wurden. Zusätzlich wird von dem Partner noch ein Stadtplanausschnitt angeboten, dessen Dienste heißen City-View und City-Server, sie werden per HTTP-Request angefordert.

Bevor nun die Web-Seite des Dienstes (s. „Abbildung 12“) angezeigt werden kann, wird von beiden Partnern geprüft, ob und welche Bilder vorhanden sind, damit der Nutzer nur Informationen anwählen kann, die auch angezeigt werden können.

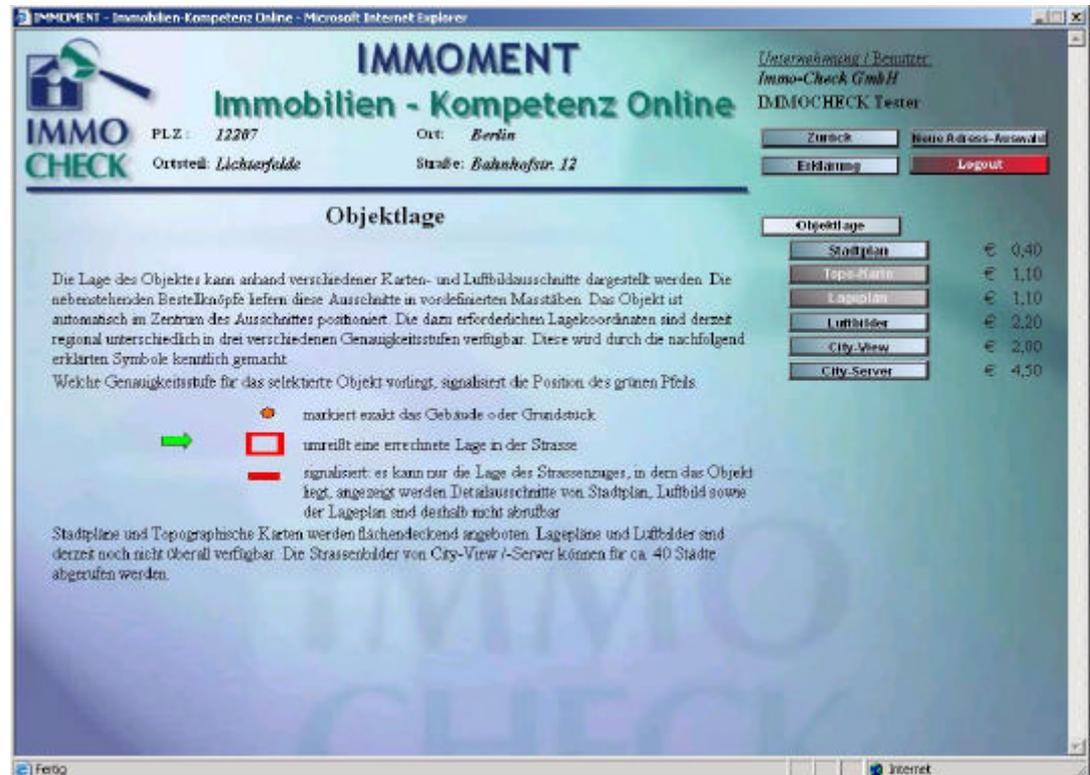


Abbildung 12 – Web-Ansicht der Dienstes Objektlage

In der „Abbildung 12“ wird im linken Teil durch einen Pfeil gekennzeichnet, in welcher Qualität die Daten vorliegen. Der Erklärungstext der Objektlage soll dies genauer erläutern:

„Die Quelle für Lage-Koordinaten, mit denen Objekte hausgenau positioniert werden können, sind die Vermessungsämter. Derzeit bieten nur die Bundesländer Nordrhein-Westfalen und Niedersachsen diese Koordinaten flächendeckend und zentral aufbereitet an. In anderen Ländern sind diese Lage-Koordinaten derzeit oft nur für einige Städte zugänglich. Darüber hinaus sind auch die Lagepunkte dieser Koordinaten trotz ihrer Präzision je nach Vermessungsamt unterschiedlich definiert: Sie können den Eingang des Objektes markieren oder sie liegen in der Mitte des Grundstückes oder des Gebäudes. Falls auf einem Grundstück mehrere Gebäude mit unterschiedlichen Hausnummern liegen, wird oft nur eine Grundstückskordinate geliefert. Diese Unterschiede sind gegebenenfalls zu beachten, können jedoch anhand der Karten mit Grundstücksgrenzen (Lagepläne) leicht erkannt werden.

Damit jedoch auch Objekte, für die noch keine hausgenauen Lage-Koordinaten verfügbar sind, in IMMOMENT so gut als möglich in Karten positioniert werden können, werden zwei weitere Koordinatenquellen genutzt: Dort, wo den Straßennetzen der Navigationssysteme bereits für die einzelnen Straßenabschnitte zwischen den Kreuzungen Hausnummernbereiche zugeordnet sind, wird die Lagekoordinate für eine Hausnummer errechnet. Dies ist in der Regel mit einer Genauigkeit von +/- 10 Meter möglich, ist jedoch sehr von der Bebauungsstruktur abhängig. Diese berechnete Koordinate markiert jedoch die Lage des Objektes schon sehr viel zutreffender, als wenn diese nur anhand des Straßenabschnittes in der Karte markiert werden kann, in dem das Objekt liegt. Diese grobe Lagemarkierung wird daher insbesondere im ländlichen Bereich außerhalb der Länder Nordrhein-Westfalen und Niedersachsen genutzt, wenn genauere Koordinaten nicht verfügbar sind.

Die in der Leitseite unterschiedenen drei Genauigkeitsstufen basieren auf dieser durch die Verfügbarkeit von Koordinaten bedingten Situation. Es wird daher automatisch angezeigt (durch den grünen Pfeil), mit welcher Genauigkeit das jeweilige Objekt in den Karten- und Luftbildausschnitten dargestellt werden kann. Um dies auch in den Kartenbildern wieder erkennen zu können, werden drei unterschiedliche Markierungssymbole verwendet. Da die Darstellung der Lage von Objekten in kleinen Ausschnitten (Lageplan und Luftbild-Detail) irreführend ist, wenn nur die Markierung des Straßenabschnittes möglich ist, in dem das Objekt liegt, werden solche Detailausschnitte bei dieser Genauigkeitsstufe automatisch nicht angeboten. Sofern für ein Objekt keine der zuvor erläuterten Koordinatenarten vorliegen, kann seine Lage in Karten nicht dargestellt werden.“

10.1 Stadt-, Lageplan, Topo-Karte und Luftbilder

Die Stadtpläne liegen flächendeckend von Deutschland vor, von mehreren Bundesländern und den meisten Städten mit mehr als 50.000 Einwohnern liegen derzeit auch Luftbilder vor. Topographische Karten und Lagepläne sind derzeit etwas seltener als die Luftbilder vom Nutzer aufrufbar. Exemplarisch wird in „Abbildung 13“ der Dienst „Luftbilder“ veranschaulicht.

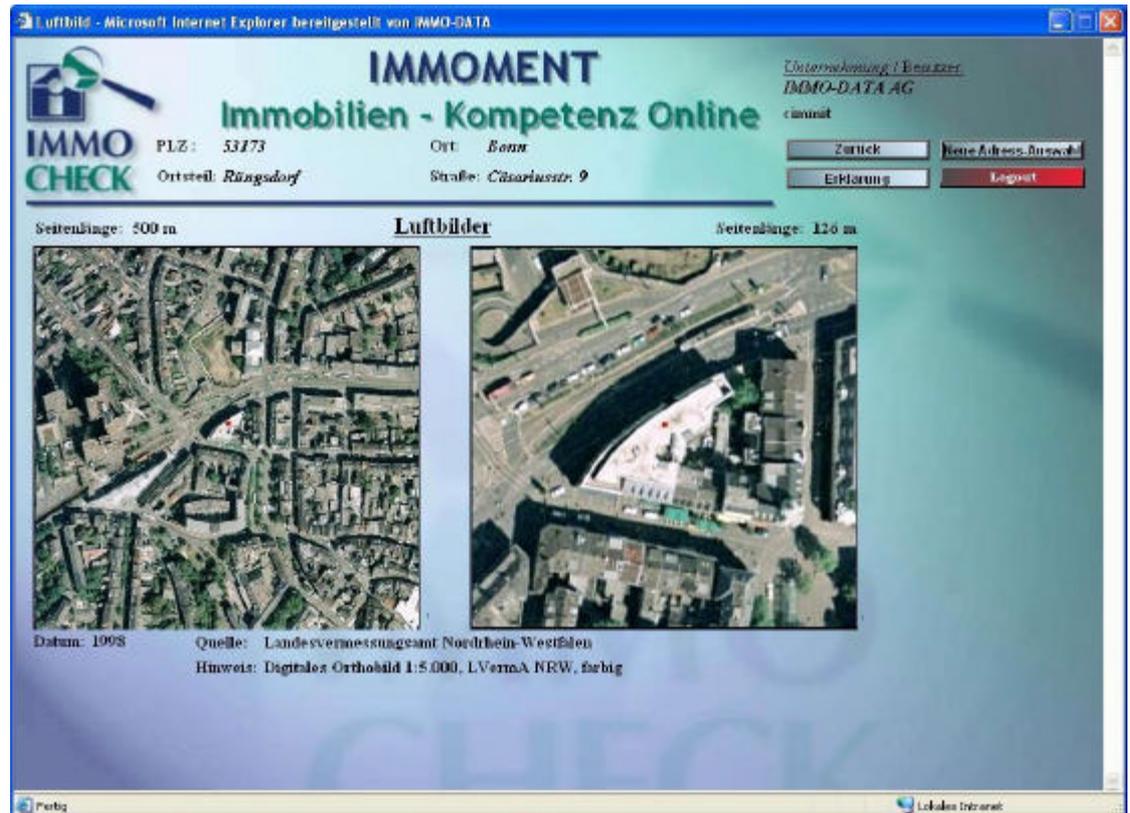


Abbildung 13 – Web-Ansicht des Dienstes Luftbilder

Bevor die Bilder dieses Partners dargestellt werden können, müssen drei Server des Partners abgefragt werden. Nach Anwahl des Dienstes „Objektlage“ wird per Web-Service der erste Server des Partners angefragt. Dieser liefert, nach Übergabe von vier Gauß-Krüger-Koordinaten, die die Eckpunkte einer Quadratfläche umspannen, eine Liste der lieferbaren Bildinformationen für diese Fläche zurück.

Exemplarisch wird nachfolgend der (gekürzte) zurückgegebene XML-Strom des DATAINFO-Service des Lieferanten (nur für die lieferbaren Luftbilder) aufgeführt:

```
<?xml version="1.0"?>
<DATAINFO>
  <REQUEST>5/9/2003 2:58:59 PM</REQUEST>
  <OUTPUT>XML</OUTPUT>
  <ITEM_LIST>
    <ITEM ID="1">
      <SOURCE_COORD_SYS>GK2</SOURCE_COORD_SYS>
      <LOGICAL_LAYER_LIST>
        <LOGICAL_LAYER ID=" 1">
          <LOGICAL_LAYER_ID>LVANRWDGK50103</LOGICAL_LAYER_ID>
          <GROUP>DOB5</GROUP>
          <PRODUCT>DOB5_01_NRW</PRODUCT>
          <CONTRACTOR>Landesvermessungsamt NRW</CONTRACTOR>
          <COMMENT>Digit. Orthobild 1:5.000, LVermA NRW, s/w </COMMENT>
          <SCALE> 5000</SCALE>
          <RESOLUTION> .3125</RESOLUTION>
          <MAP_RANGE_MIN>1</MAP_RANGE_MIN>
          <MAP_RANGE_MAX>1000</MAP_RANGE_MAX>
        </LOGICAL_LAYER>
        <LOGICAL_LAYER ID=" 2">
          <LOGICAL_LAYER_ID>LVANRWDOB52401</LOGICAL_LAYER_ID>
          <GROUP>DOB5</GROUP>
          <PRODUCT>DOB5_24_NRW</PRODUCT>
          <CONTRACTOR>Landesvermessungsamt NRW</CONTRACTOR>
          <COMMENT>Digit. Orthobild 1:5.000, LVermA NRW, farbig</COMMENT>
          <SCALE> 5000</SCALE>
          <RESOLUTION> .3175</RESOLUTION>
          <MAP_RANGE_MIN>1</MAP_RANGE_MIN>
          <MAP_RANGE_MAX>5000</MAP_RANGE_MAX>
        </LOGICAL_LAYER>
        <LOGICAL_LAYER ID=" 4">
      </LOGICAL_LAYER_LIST>
    </ITEM>
  </ITEM_LIST>
</DATAINFO>
```

Der DATAINFO-Service des Partners liefert alle lieferbaren Bilder, in Gruppen (GROUP – Luftbilder haben die Kennungen „DOB5“, bzw. „LBK“) und Untergruppen (PRODUCT) aufgeteilt. Die Angaben, von welchem Landesvermessungsamt (LOGICAL_LAYER_ID, bzw. CONTRACTOR) dieses Bild geliefert wird und in welcher Qualität (SCALE, RESOLUTION, bzw. in Textform: COMMENT), muss nun von der Web-Applikation IMMOMENT ausgewertet werden. Dabei werden in Priorität farbige Bilder mit dem genauesten Kartenausschnitt (SCALE) und der besten Auflösung (RESOLUTION) angezeigt. Im obigen Beispiel würde die farbige Karte (ID=" 2") angezeigt, obschon die andere Karte eine etwas bessere Auflösung aufweist.

Nach Auswertung der Liste können vom Nutzer nur die Button angewählt werden, für die auch Bilder vom Lieferanten geliefert werden können, die übrigen Button werden grau dargestellt (s. „Abbildung 12“ – Topo-Karte und Lageplan konnte nicht geliefert werden).

Nachdem einer der Dienste angewählt wurde, z.B. der Dienst „Luftbilder“, wird beim ersten Aufruf von einem weiteren Server des Partners das Jahr angefordert, in welchem die Bilder erstellt wurden. Das Jahr wird dann als Datum unten links im Browser-Fenster angezeigt (s. „Abbildung 13“). Bei der ersten Abfrage eines Bildes werden für alle vier Bildarten das Erstellungsjahr vom DATEINFO-Service angefordert, wodurch die weiteren Bildabfragen schneller durchgeführt werden, da der DATEINFO-Service nicht mehr angefordert wird. Der DATEINFO-Service stellt den zweiten Server-Aufruf dieses Partners dar.

Damit nun das Bild angezeigt werden kann, muss noch ein dritter Service des Partners angefordert werden, der MAP-Service. Der MAP-Service erstellt nach bestimmten Vorgaben¹²², die ihm übergeben werden müssen, eine JPEG-Datei, welche in einem virtuellen Verzeichnis abgelegt wird. Die Rückgabe des MAP-Service ist ein XML-Strom, der im Grunde nur die URL enthält, unter der die erstellte JPEG-Datei angefordert werden kann. Die URL-Rückgabe sieht z.B. wie folgt aus:

195.27.54.40/TMSWS4/Cache/{9C31684D-6C36-49DB-A777-7CA56C797E7C}.jpg

In diesem virtuellen Verzeichnis liegt das Bild fünf Minuten vor und wird dann gelöscht.

¹²² **Vorgaben für den MAP-Service** sind: 1. Landkartenausschnitt, begrenzt durch vier Gauß-Krüger Koordinaten, die die vier Eckpunkte der Landkarte angeben, 2. Bildausschnittgröße, wie sie im Browserfenster angezeigt wird, 3. spezielle Darstellung der Koordinatengenauigkeit (z.B. Punkt).

10.2 City-View

Neben den gerade beschriebenen Bildinformationen direkt zur Immobilie runden die Straßenbilder den Gesamteindruck der Lage ab. Der Dienst „City-View“ stellt dabei die statische Ansicht der verfügbaren (sechs bzw. acht) Kamera-Bilder dar. Über die Lage-Koordinaten werden die Bilder so nah wie möglich vor der angegebenen Adresse angezeigt. Ob Daten für die gesuchte Adresse vorhanden sind enthält die Tabelle „im_staedte_info“ (s. Kapitel „7.2.2“). Bei Erstellung des Pflichtenheft war für die Anzeige der Bilder ein ActiveX nötig, das ist aus sicherheitstechnischer Sicht erfreulicherweise (s. „1.4 Trennung von Daten und GUI“) nicht mehr erforderlich, da der Partner die Bilder mittlerweile auch im JPEG-Format liefert.

10.3 City-Server

Startend mit der City-View-Ansicht ermöglicht der City-Server in zehn Meter Schritten in Fahrtrichtung (bzw. gegen die Fahrtrichtung) am Objekt vorbei zu „fahren“. Das ermöglicht einen genaueren Eindruck der Wohnlage.

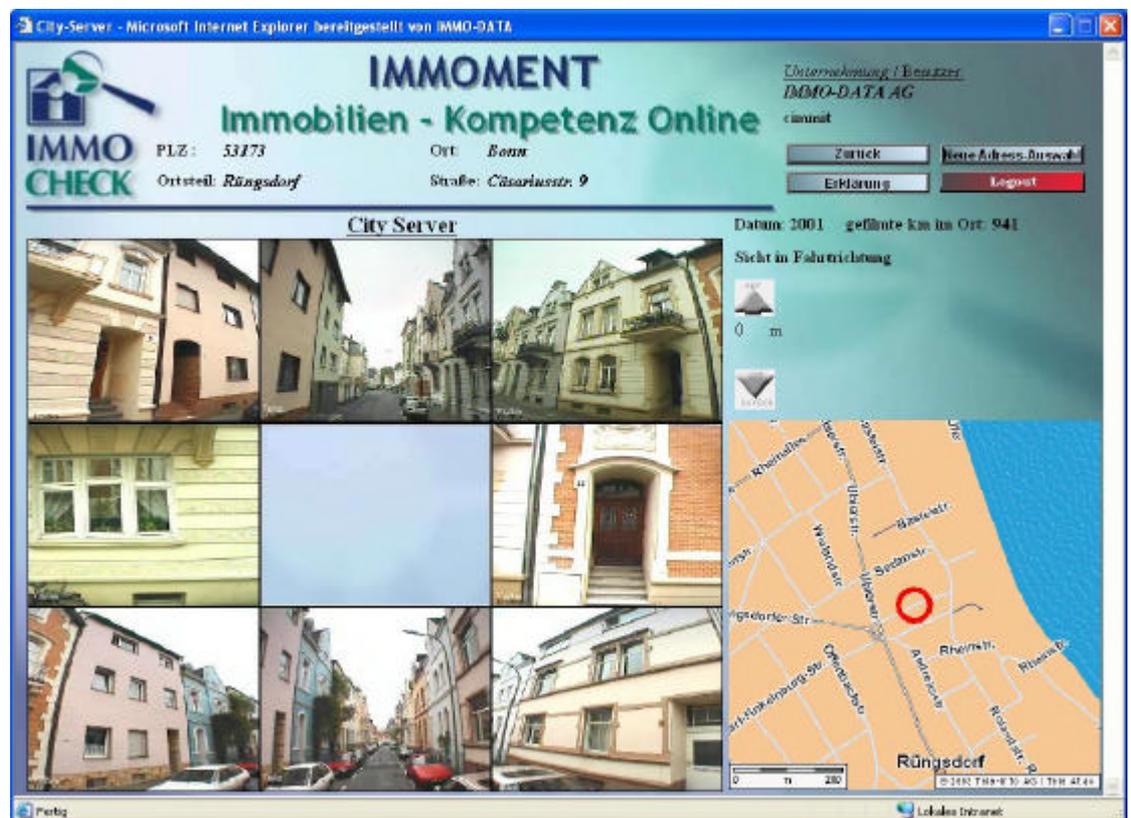


Abbildung 14 – Web-Ansicht des Dienstes City-Server

11 Web-Service Aufruf

Die Web-Applikation IMMOMENT kann neben der direkten Dialogsteuerung per Internet-Browser auch Daten per Web-Services in einem XML-Strom zurückliefern, damit die so übermittelten Daten in eigene Anwendungen des Kunden integriert werden können. Es stehen derzeit vier Web-Services zur Verfügung, beispielhaft werden die Schnittstellen des Web-Services – ImmoAgentAuthenticate – beschrieben.

ImmoAgentAuthenticate stellt die User-Prüfung dar, sie wird grundsätzlich von allen Web-Services aufgerufen, kann aber wie folgt beschrieben auch direkt aufgerufen werden.

11.1 ImmoAgentAuthenticate Aufruf

ImmoAgentAuthenticate(string xml_str)

Der zu übergebende string (xml_str) ist ein XML-Strom und hat folgenden Aufbau:

```
<immoment>
  <user>
    <key>...</key>
    <password>...</password>
    <name_neu />
    <key_neu />
    <password_neu />
    <kostenstelle_neu />
  </user>
</immoment>
```

Der angefragte User muss bereits in der User-Datenbank von IMMOMENT angelegt sein, eine Neuanlage ist nicht möglich, diese muss erst bei der IMMO-DATA angemeldet werden. Es werden Firmen aber grundsätzlich mehrere Usereinträge zur Verfügung gestellt, die dann durch die Felder `<name_neu>`, `<key_neu>`, `<password_neu>` und `<kostenstelle_neu>` angepasst werden können.

Die Felder `<key>` und `<password>` sind Pflichtangaben, die weiteren Felder sind optional. Optionale TAGs (z.B. `<name_neu>`) können weggelassen werden bzw. werden nur benötigt, wenn die bisher erfassten Daten geändert werden sollen:

- Ein Eintrag in `<name_neu>` ändert den bisherigen Namenseintrag
- Ein Eintrag in `<key_neu>` ändert die bisherige Benutzerkennung
- Ein Eintrag in `<password_neu>` ändert das bisherige Passwort
- Ein Eintrag in `<kostenstelle_neu>` ändert die bisherige Kostenstelle

11.2 ImmoAgentAuthenticate Rückgabe

Zurückgegeben wird wieder der TAG `<user>` mit dem Child `<key>`. Es wird ein weiterer TAG `<error>` mit dem Unterknoten `<user>` und den weiteren Unterknoten `<errorcode>` und `<errmsg>` zurückgegeben. Ist der `<errorcode>` „0“, dann ist alles in Ordnung, die genauere Erläuterung wird in der `<errmsg>` gegeben.

Zusätzlich wird an das root-TAG `<immoment>` das Attribut „date_time“ angehängt, das das Datum (4j-mm-tt) und die Uhrzeit der Rückgabe minutengenau angibt.

Die übliche Rückgabe:

```
<immoment date_time="2003-03-15 16:00">
  <user>
    <key>...</key>
  </user>
  <error>
    <user>
      <errorcode>0</errorcode>
      <errmsg>User ist berechtigt!</errmsg>
    </user>
  </error>
</immoment>
```

Mit den gewünschten Änderungen `<name_neu>` `<key_neu>` `<password_neu>` und `<kostenstelle_neu>` wird folgendes zurückgegeben:

```
<immoment date_time="2003-03-15 16:00">
  <user>
    <key>aaaaaa1</key>
  </user>
  <error>
    <user>
      <errorcode>0</errorcode>
      <errmsg>user - Name, Benutzer, Passwort,
        Kostenstelle wurde geändert!</errmsg>
    </user>
  </error>
</immoment>
```

Ist der angefragte Nutzer nicht angelegt, wird folgendes zurückgegeben:

```
<immoment date_time="2003-03-15 16:00">
  <user>
    <key>aaaaaa1</key>
  </user>
  <error>
    <user>
      <errorcode>1</errorcode>
      <errmsg>User ist unbekannt!</errmsg>
    </user>
  </error>
</immoment>
```

12 Schlussbemerkung

Mit der vorliegenden Diplomarbeit sollte nicht nur dargestellt werden in welchen Schritten die Web-Applikation IMMOMENT erstellt wurde, sondern es sollten auch die Probleme aufgeführt werden, die bei der Erstellung auftraten und bewältigt werden mussten. Die Probleme waren hauptsächlich darin zu finden, dass mehrere Partner nicht nur verschiedenartige Daten zur Verfügung stellten, sondern dass diese Daten auch auf verschiedenen Wegen in IMMOMENT integriert werden mussten. Wie bereits im „Abstrakt“ erwähnt, ergaben sich auch einige Probleme durch die Umsetzung als Web-Applikation.

12.1 Ziele

Es sind noch nicht alle Ziele erreicht worden, zwar ist die technische Umsetzung bis auf wenige Punkte, die im folgenden Unterkapitel „Ausblick“ aufgeführt werden, trotz der Probleme vollendet, aber das wirtschaftliche Ziel, viele Kunden zu gewinnen, ist noch nicht erreicht. IMMOMENT „lebt“ durch die Partnerdaten, das hat zur Folge, dass zwar viele Interessenten überrascht sind, welche Datenvielfalt sie zu einer eingegebenen Anschrift vorfinden, aber die Qualität, bzw. Vielfalt der Informationen ist derzeit nur in wenigen Bundesländern sehr hoch. Aus diesem Grunde warten einige Interessenten ab, in der Hoffnung, dass sie auch in den Regionen, die sie bevorzugt abfragen werden, eine höhere Qualität vorfinden. Das wirtschaftliche Ziel war zwar nicht Thema dieser Arbeit und soll nur am Rande erwähnt werden, es zeigt aber, dass die Qualität der Web-Applikation aus Sicht der Kunden entscheidend von den beteiligten Partnern abhängt.

12.2 Ausblick

Wie bereits im Pflichtenheft erwähnt (Anhang A – 1.2 Wunsch- oder Kann-Kriterien), ist auch weiterhin die Umsetzung von IMMOMENT auf Handheld / PDA-Computern bzw. auf einem Handy geplant.

Der Dienst „Marktdaten“ wird deutlich erweitert, das kündigte der Partner bereits an. Die zusätzlichen Darstellungen müssen allerdings noch abgesprochen und die neuen Daten in die Datenbank integriert werden.

Weitere Partner sollen hinzukommen, ein Projekt zeigt in Straßenkarten weitere Merkmale an, z.B. wurden direkt in der Karte Infrastrukturmerkmale hervorgehoben, wie Bundes-Bahn- oder U-Bahn-Haltestellen. Es wurden auch farblich in der Straßenkarte die Wohnobjekte mit einer mittleren Miete anders dargestellt als Wohnobjekte mit gehobener Miete. Dieses Straßenkartensystem soll in naher Zukunft in IMMOMENT integriert werden.

Mehrsprachigkeit ist ebenfalls ein geplanter Punkt. Dafür wird die Tabelle „im_hilfe_texte“ um das Attribut Sprache erweitert, das die Landesprache-Kürzel „de“ für Deutsch oder „en“ für Englisch enthält (bzw. für weitere Sprachen). In der entsprechenden Sprache müssen dann selbstverständlich noch die Texte erfasst werden. Wählt ein Nutzer aus dem englischen Sprachraum die Anwendung an, wird von der Web-Applikation die Spracheinstellung des Client-Browser erkannt und aus der Tabelle der entsprechende Text angezeigt.

Die Web-Services werden ausgebaut, insbesondere wenn weitere Daten durch neue Partner hinzukommen. Die Web-Services sollen dabei eine „Fat Client“ Architektur unterstützen, wodurch die von IMMOMENT gelieferten Daten weiter genutzt werden können. Die bestehenden Client-Server-Produkte der IMMO-DATA AG sind bereits teilweise in der Lage nach einer Adresseingabe bestimmte Daten aus IMMOMENT anzufordern um diese z.B. für Plausibilitätsprüfungen zu nutzen. Somit bereichert IMMOMENT andere Produkte der IMMO-DATA AG durch zusätzliche Informationen.

Des Weiteren wird auch noch der Dienst „Wirtschaftsauskunft“ umgesetzt. Dieser wird Nutzern, die allerdings ein berechtigtes Interesse vorweisen müssen, über die finanzielle Situation von Firmen in einem Immobilien-Objekt Auskunft geben.

Anhang A – Pflichtenheft

Die Software-Entwicklung wird (nach Balzert) in sechs Phasen aufgeteilt:

- Planungsphase
- Definitionsphase
- Entwurfsphase
- Implementierungsphase
- Abnahme-/Einführungsphase
- Wartungs-/Pflegephase

Dabei stellt das Ergebnisdokument der Definitionsphase das Pflichtenheft dar. Das Pflichtenheft wird vom Auftraggeber erstellt und enthält die fachlichen Anforderungen an das Software-Produkt aus Sicht des Auftraggebers. Es sollte grundsätzlich nur das „Was“ behandelt werden und nicht das „Wie“.

1. Zielbestimmung

1.1 Musskriterien

Die Web-Applikation IMMOMENT muss als B2B¹²³-Lösung auf drei Arten Informationen zu einem Objekt an den Anwender übermitteln:

1. Per Internet-Online-Verbindung und Anwahl mit einem Browser durch Benutzer- und Passwort-Eingabe. Dabei wählt der Anwender jede Information selbst aus, sowohl die Textdaten als auch die Bildinformationen werden ihm im Browser direkt angezeigt.
2. Zu einem bestimmten Objekt kann der Anwender aus den bereits installierten IMMOBÜRO Anwendungen direkt in die IMMOMENT Web-Applikation wechseln, ohne sich anmelden zu müssen. Er muss dann wie unter 1. Online die Informationen anwählen und bekommt diese direkt angezeigt.

¹²³ B2B – Business-to-Business: Transaktionen zwischen zwei Unternehmen – bedeutet, dass IMMOMENT nicht dem Endkunden, sondern nur Sachbearbeitern von Firmen zugänglich ist.

3. Zu einem oder mehreren Objekten müssen die Informationen aus IMMOMENT direkt in den bereits vorhandenen Datensatz der IMMOBÜRO Anwendung übernommen werden. Dabei müssen die Daten „automatisch“ ohne Browseransteuerung der Internetseiten in die IMMOBÜRO-Umgebung integriert werden.

Weitere Voraussetzungen sind:

- In der Kundenverwaltung muss über Berechtigungen geregelt sein, welche Informationen oder Dienste (siehe Produkt-Leistungen) der Benutzer auswählen darf.
- Jede im Browser ausgewählte Information darf erst durch die Anwahl (Buttondruck) und nach Prüfung Kosten verursachen (pay per Click – pay on demand).
- Die Kosten müssen getrennt verwaltet werden, damit Rechnungen nach Benutzer und Kostenstellen gestaffelt an die Firma ausgegeben werden können.
- Die Anwendung muss über eine gesicherte Internet-Verbindung (https) verfügen.
- IMMOMENT muss in standardisierter Form Karten- und Luftbild-Ausschnitte in jeweils geeigneten Maßstäben liefern, in denen das Objekt bereits automatisch zentriert und markiert ist.
- IMMOMENT sollte mausgesteuert, intuitiv und mit wenigen „Clicks“ benutzt werden können, damit keine langen Einarbeitungszeiten nötig werden.

1.2 Wunsch- oder Kann-Kriterien

In einer späteren Phase ist es wünschenswert, dass die Web-Applikation IMMOMENT auch über ein Handheld / PDA-Computer oder per Handy über WAP oder UMTS-Verbindung aufgerufen werden kann.

1.3 Abgrenzungskriterien

- Beliebige Browser (insbesondere ältere Versionen) können nicht unterstützt werden (siehe Software).
- Die Web-Applikation IMMOMENT wird durch die Daten der Partner der IMMO-CHECK GmbH gebildet, aus diesem Grunde wird es nötig sein, dass zur Nutzung mancher Daten zusätzliche Programme auf dem Client des Anwenders installiert werden müssen (siehe Software).

2. Produkt-Einsatz

2.1 Anwendungsbereiche

Online sind sofort Informationen über den Wert eines Objektes und über sein Umfeld abrufbar. Dadurch wird einem Sachbearbeiter (siehe Zielgruppe) bereits im Gespräch mit einem Kunden ermöglicht, direkt das Objekt zu beurteilen.

2.2 Zielgruppen

IMMOMENT ist bestimmt für den Einsatz von Sachberatern einer Bank, Bausparkasse, Versicherung und von Anwendern der Immobilienwirtschaft, die den Wert eines Objektes ermitteln müssen und entsprechend genaueste Informationen über dieses Objekt und sein Umfeld benötigen.

2.3 Betriebsbedingungen

Da es sich um eine Web-Anwendung handelt, sollte sie rund um die Uhr, also im Dauerbetrieb laufen. Somit muss aber auch gewährleistet werden, dass bei Störungen des Systems, sei es durch Probleme der Hardware oder Software, es innerhalb von wenigen Stunden wieder in einen „aktiven“ Status versetzt wird. Das wird durch den Einsatz mehrerer Web- und Datenbank-Server in einem großen Rechenzentrum gewährleistet.

3. Produkt-Umgebung

3.1 Software

Da es sich um eine Client-Server-Architektur handelt, muss dieser Punkt aus zwei Sichten beantwortet werden.

1. Der Server-Computer benötigt:
 - Windows 2000 oder Windows XP
 - Microsoft .Net Framework 1.0 Version 1.0.3705
 - Microsoft SQL-Server ab Version 7.0
 - Microsoft Internet-Information-Server IIS
2. Der Client-Computer, der dem Anwender ermöglicht die Web-Applikation aufzurufen, benötigt:
 - das Betriebssystem beim Client kann Windows ab der Version 98, Unix oder Linux sein. Voraussetzung ist, dass ein Internet-Browser und die im Folgenden aufgeführten Programme auf dem Betriebssystem installiert werden können.
 - Internet-Browser (Internet-Explorer ab Version 5.0 o. Netscape ab Version 6.1).
 - Pdf-Reader, das Programm zeigt Texte in speziell gespeicherter Form an, so genannte pdf-Dateien – kostenlos erhältlich unter www.adobe.de.
 - Eine Bildanzeige-Software – erhältlich unter <http://62.132.30.76/lura.html>.
 - Manche Dateien, z.B. Hilfetexte, können heruntergeladen und danach Offline gelesen werden. Dazu wird ein Programm benötigt, das komprimierte Dateien wieder entpackt. Gebräuchlich ist „WinZip“ – erhältlich unter www.winzip.de.

3.2 Hardware

Auch dieser Punkt wird aus zwei Sichten beantwortet:

1. Der Server-Rechner benötigt einen PC, der in der Lage ist die unter Software aufgeführten Programme auszuführen. Die Anforderungen an den Server sind weder in der Performance noch in der Datenmenge überdurchschnittlich hoch, so dass auch andere Applikationen parallel von dem System verwaltet werden können. Der Web-Server (.Net Framework) und der Datenbank-Server (MS-SQL Server) sollten auf getrennten Computern ihre Dienste anbieten. Die

Web-Applikation ist auf verschiedenen Webservern, z.B. in einer Webfarm¹²⁴ im Load Balancing¹²⁵ lauffähig und kann somit auf mehreren Computern installiert werden. Werden alle Server auf einem Computer installiert, so sollte die Festplattenkapazität wenigstens zehn Gigabyte umfassen.

2. Der Client-Rechner muss ebenfalls in der Lage sein, die unter Software aufgeführten Programme auszuführen. Des Weiteren muss dieser über einen Internet-Zugang verfügen. Da einige Bilddaten übertragen werden, ist eine schnelle Internet-Datenleitung von Vorteil (ISDN oder DSL), aber nicht ausdrücklich nötig. Der angeschlossene Monitor muss in der Lage sein eine Auflösung von 1024 x 768 Pixel darstellen zu können, da die Web-Applikation für diese Auflösung optimiert wurde.

3.3 Produktschnittstellen

Die Bilddateien werden direkt von den Partner-Servern geladen und angezeigt. Diese werden also nicht lokal auf dem Server gespeichert, somit fällt kein zusätzlicher Verwaltungsaufwand (z.B. für temporäre Dateien) auf dem Server an. Die Textdaten der Partner-Server werden über XML¹²⁶-Schnittstellen in IMMOMENT integriert.

Sowohl die Bild- als auch die Text-Informationen können aus IMMOMENT über spezielle von .Net Framework angebotene Web-Services¹²⁷ aufgerufen und somit in Fremdanwendungen integriert werden.

4. Produkt-Funktionalität

Als Online-Portal stellt IMMOMENT Daten und Informationen der Partner-Server zur Verfügung.

¹²⁴ Bei einer Webfarm werden statt eines Einzigen eine Reihe von Webservern verwendet, die alle die gleichen Aufgaben übernehmen und sich gegenüber dem Besucher als ein Websystem präsentieren.

¹²⁵ Load Balancing Systeme verteilen die Last zwischen Geräten (Servern) in einer Web-Umgebung

¹²⁶ XML steht für "Extensible Markup Language", zu deutsch "erweiterbare Auszeichnungssprache". Sie wurde entworfen, um die Funktionalität des Web zu erweitern, indem man eine flexiblere und anpassungsfähige Identifizierung von Informationen ermöglicht. XML ist ein selbsterklärendes Dateiformat mit eigener Struktur. Somit ist es hervorragend geeignet um Textinformationen im Internet auszutauschen und weiterzuverarbeiten.

¹²⁷ Web-Services stellen unabhängige, selbsterklärende und modulare Applikationen dar, die im Internet publiziert, lokalisiert und aufgerufen werden können. Web Services führen Funktionen aus - von simplen Anfragen bis zu komplizierten Geschäftsprozessen. Ist ein Web Service implementiert, können andere Applikationen (auch andere Web Services) diesen erkennen und aufrufen. Web Services nutzen verschiedene Standard-Protokolle, darunter: Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP) sowie Universal Description, Discovery and Integration (UDDI).

Zuerst wird eine Zugangs- und Adressprüfung durchlaufen. Die folgende Abbildung (1) zeigt das ER-Modell der Tabellen der Datenbank im_basis. Die zentrale Tabelle ist die Benutzertabelle im_user_datan, diese enthält hauptsächlich Zugangsinformationen.

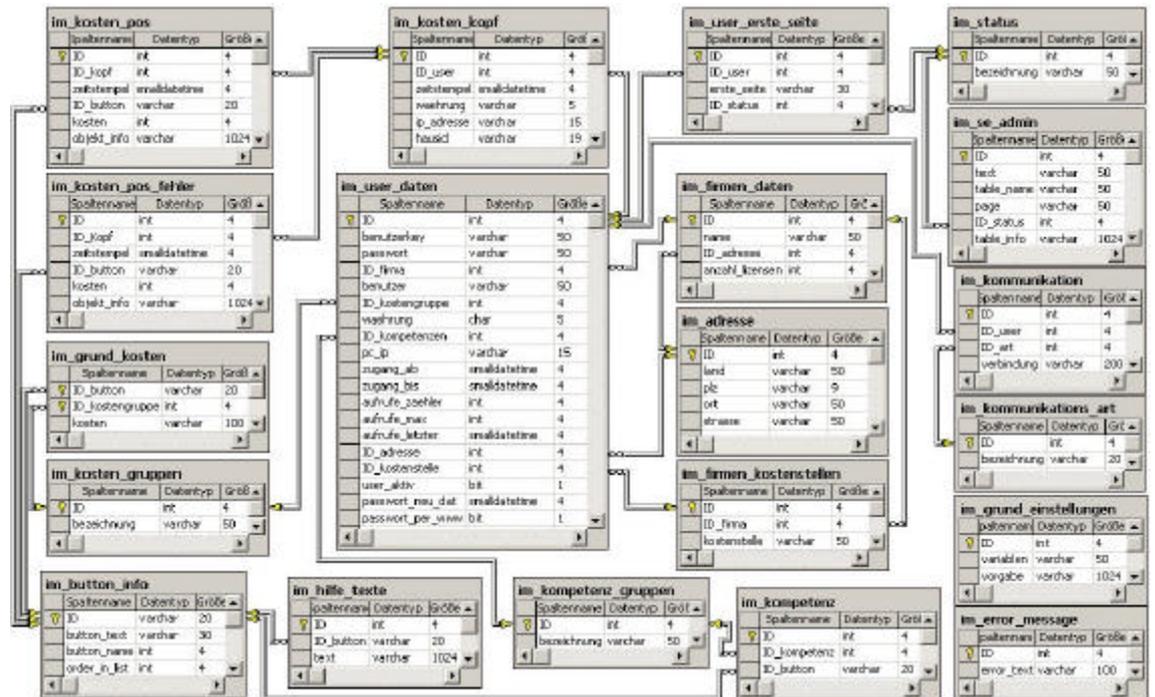


Abb. 1)

a) Zugang zu IMMOMENT

Die Authentizitätsprüfung stellt fest, ob die eingegebene Benutzerkennung und das Passwort in der Benutzer-Tabelle (im_user_datan - der Datenbank im_basis) vorhanden sind. Eine darauffolgende Prüfung erkennt, ob trotz der vorhandenen Benutzerkennung/Passwort Kombination der Zugriff verweigert werden muss, wenn:

- das aktuelle Datum noch nicht den Wert zugang_ab erreicht hat,
- das aktuelle Datum den Wert zugang_bis überschreitet,
- die IP-Adresse des sich gerade angemeldeten Benutzers unterschiedlich zu dem Wert pc_ip ist. Diese Prüfung wird nur vorgenommen, wenn ein Wert bei pc_ip eingetragen wurde, da nicht alle Kunden über eine feste IP-Adresse verfügen.
- der Benutzer versucht sich häufiger anzumelden, als der Eintrag in dem Wert aufrufe_max zulässt. Alle Anmeldungen werden in dem Wert aufrufe_zaeher gespeichert. Diese Prüfung wird nur vorgenommen, wenn ein Wert bei aufrufe_max eingetragen wurde.

b) Objektdaten – Adressprüfung und Georeferenzierung

Nach Eingabe der Adresse wird in einem ersten Schritt von einem Partner geprüft, ob die Kombination Postleitzahl, Ort und Straße in einer Datenbank vorhanden ist, die alle Objekte Deutschlands enthält. Konnte die eingegebene Adresse in der Datenbank nicht gefunden werden, sucht ein Algorithmus nach ähnlichen Schreibweisen der Straße und des Ortes und es wird dem Benutzer möglicherweise eine Liste von ähnlich geschriebenen Adressen zur Auswahl am Bildschirm angezeigt. Der Benutzer kann nun eine Adresse aus der Liste auswählen.

Konnte der Vergleichs-Algorithmus keine ähnlichen Einträge finden, so wird dem Benutzer eine entsprechende Meldung angezeigt und er muss nun selbst die Schreibweise korrigieren und eine neue Prüfung auslösen.

Wurde die Adresse gefunden, so wird sie georeferenziert, das bedeutet, dass der angegebenen Adresse eine 18-, bzw. 19-stellige alphanumerische Kennung (die Haus-ID) zugewiesen wird, die für jedes Objekt in Deutschland eindeutig ist.

Diese Kennung wird benötigt, um auf die Daten der objektbezogenen Tabellen zugreifen zu können. An der Adresse Oxfordstr. 2a in 53111 Bonn kann der Aufbau der Haus-ID 053140000195400002 exemplarisch erläutert werden:

Stellen 1-2	(05)	Kennung für das Bundesland (Nordrhein-Westfalen)
3-5	(314)	Kennung für den Landkreis (Bonn)
6-8	(000)	Kennung für die Gemeinde (kein Eintrag somit ist die Gemeinde gleich dem Landkreis also Bonn)
9-13	(01954)	Kennung für die Straße (Oxfordstraße)
14-19	(00002a)	Kennung für die Hausnummer (2) mit Zusatz (a)

Erst in einem zweiten Schritt wird geprüft, ob die angegebene Hausnummer in der geprüften Straße vorhanden ist. Wird festgestellt, dass es keine Adresse mit dieser Hausnummer gibt, so kann es sich um einen Neubau handeln. Das wird dem Benutzer mitgeteilt, worauf dieser nun prüfen muss, ob eine falsche Hausnummer eingegeben wurde oder ob das Objekt der Datenbank nicht bekannt ist.

Nach der Adressprüfung und der Georeferenzierung wird nun die Kompetenzprüfung des Benutzers durchgeführt. Die Kompetenzprüfung ermittelt, welche Dienste der Benutzer aufrufen darf. Jedem Benutzer wird eine bestimmten Kompetenzstufe zugeordnet und entsprechend können nun die erlaubten Dienste aufgerufen werden. Folgende Dienste sind derzeit anwählbar (Abbildung 2 zeigt die Webseitenansicht nach der Adressprüfung mit der höchsten Kompetenzstufe):

Abb. 2)

1. Objektmfeld¹²⁸

Die in Textform gelieferten Informationen unterstützen die Beurteilung des abgefragten Objektes, seiner Nutzung, Lage und seines Umfeldes. Diese Angaben basieren auf der Auswertung der mikrogeographischen Datenbank eines Partners. Die jeweils spezifisch auf das Objekt oder sein Umfeld zutreffenden Angaben sind in einen Text eingebunden und werden nicht besonders hervorgehoben. Die dabei verwendeten absoluten Angaben, Indikatoren, Kennziffern oder Textelemente werden in einer Datenbank auf der Ebene der ca. 17,3 Mio. Objekte sowie auf der Ebene von ca. 1,4 Mio. Straßenabschnitten, in denen jeweils das Objekt liegt, verwaltet und über die Haus-ID mit der eingegebenen Anschrift verknüpft.

Auf diese Weise können auch für Hausnummern von Objekten, für die in der Datenbank keine hausspezifischen Angaben vorliegen – in der Regel bei Neubauten –, dennoch die für den Straßenabschnitt gültigen Angaben ausgegeben und genutzt werden.

¹²⁸ Der Text zum Objektmfeld wurde primär der Hilfeseite der Web-Applikation entnommen und ergänzt. Der Text der Hilfeseite wurde in Zusammenarbeit der Fa. Microm und IMMO-DATA AG erstellt

Bei der Verwendung dieser Angaben ist zu beachten, dass es sich hierbei nicht um Angaben handelt, die aus einer Begehung konkret vor Ort gewonnen wurden. Die Angaben basieren auf einer mikrogeographischen Datenbank, in der Daten aus verschiedensten maschinell erschließbaren Quellen insbesondere zu den dort lebenden Konsumenten und den dort ansässigen Firmen unter Beachtung der Auflagen des Datenschutzes in einer speziellen Form, die an anderer Stelle erläutert wird, aufbereitet sind. Insbesondere wegen der mit zum Teil komplexen statistischen Verfahren (z.B. Clusterung) durchgeführten Aufbereitung von vielen Mio. Einzeldaten müssen „Unschärfen“ im konkreten Einzelfall hingenommen werden. Der Nutzen mikrogeographischer Daten besteht primär in der Möglichkeit, auch dann, wenn konkrete, personenbezogene Angaben nicht vorliegen, die Konsumentenstruktur in den Objekten qualitativ im Rahmen statistischer Wahrscheinlichkeitstoleranzen beschreiben zu können. Die Daten werden im Turnus von etwa zwei Jahren komplett aktualisiert, Änderungen für verschiedene Bundesländer finden teilweise halbjährlich statt.

- Zunächst werden Angaben zur Größe des Hauses sowie zur Zahl der Haushalte und Firmen als Nutzer im Haus gemacht.
- Die folgenden Angaben beziehen sich primär auf den Straßenabschnitt, in dem das Objekt liegt. Bei kleinen Straßen ist dies die Straße insgesamt. Längere Straßen sind durch Hausnummernbereiche unterteilt. Häufig wird dabei auch die linke o. rechte Straßenseite unterschieden. Es wird angegeben, wie viele Häuser dem Abschnitt zugeordnet sind und wie diese sich nach Hausgrößen verteilen.
- Die wesentliche Beschreibung der im Objekt selbst und im Straßenabschnitt wohnenden Konsumenten wird in diesem Abschnitt geliefert.
- Um die Zusammensetzung der Konsumenten-Typen, die in dem Straßenabschnitt wohnen, anschaulich darzustellen und zu bewerten, werden die drei wichtigsten Prozentanteile genannt und jeweils zum Vergleich der entsprechende Wert der Stadt oder des Kreises, in dem die Straße liegt, ausgewiesen.

- Ergänzend werden Angaben zum „Status“ der Bewohner gemacht, sowohl im Objekt wie auch im Straßenabschnitt. „Status“ hat sich als aussagekräftige Variable erwiesen, die insbesondere durch berufliche Qualifikation und Kaufkraft geprägt ist. Bei der Variablen „Ausländeranteil“ ist zu beachten, dass dieser Indikator sich primär auf Namensanalysen stützt. Die angegebene „Kaufkraftklasse“ bezieht sich nicht nur auf den ansonsten betrachteten Straßenabschnitt, sondern auf das zugeordnete Wohnquartier. Dies umfasst mehrere Straßenabschnitte und es leben durchschnittlich etwa 500 Haushalte in einer solchen Zelle.
- Regionalangaben schließen den Text ab. Hier werden Angaben aufgelistet, die für den Ort oder die Stadt insgesamt gelten.

2. Objektlage

Die Objektlage liefert zwei Arten von Bildern:

- a) Die Lage des Objektes kann anhand verschiedener Karten- und Luftbildausschnitte (zwei Stadtplan-Ausschnitte, topographische Karte, Lageplan, zwei Luftbilder) dargestellt werden. Die angezeigten Bildausschnitte werden in vordefinierten Maßstäben dargestellt. Das Objekt ist automatisch im Zentrum des Ausschnittes positioniert. Die dazu erforderlichen Lage-Koordinaten sind derzeit regional unterschiedlich in drei verschiedenen Genauigkeitsstufen verfügbar:
 - Ein Punkt markiert exakt das Gebäude oder Grundstück.
 - Ein Quadrat umreißt eine errechnete Lage in der Straße.
 - Eine Linie signalisiert: es kann nur die Lage des Straßenzuges, in dem das Objekt liegt, angezeigt werden. Detailausschnitte von Stadtplan, Luftbild sowie der Lageplan sind deshalb nicht abrufbar.
- b) Der City-View liefert sechs bzw. acht Bilder, die die Straße, in der das Objekt liegt, zeigen. Der City-Server stellt die gleichen Bilder wie der City-View zur Verfügung. Dieser Dienst bietet zusätzlich die Möglichkeit weitere Bilder in Zehn-Meter-Schritten am Objekt vorbei die Straße entlang vor bzw. zurück anzuschauen. Diese Bilder liegen derzeit für 43 Städte in deutschen Ballungsgebieten vor. Der Partner erweitert sein Angebot stetig.

3. Infrastruktur

Derzeit gibt es noch keine Daten für diesen Punkt.

4. Objektwert

Es wird auf Tabellen der Verkehrs- und Beleihungswertermittlung zugegriffen, die von Kunden zur Verfügung gestellt werden.

Die Objektwerte enthalten detaillierte Angaben über das Objekt, wie z.B.:

- Größe der Wohn- oder Gewerbefläche und Baujahr,
- stichwortartige Grundstücks- und Objektbeschreibungen,
- Grundbuchdaten,
- Bauwert (aufgeteilt nach den einzelnen Gebäudeteilen),
- Sachwert,
- Ertragswerte (Rohertrag, Bewirtschaftungskosten, Bodenwertverzinsung, Gebäudereinertrag),
- den aus den obigen Werten resultierende Beleihungs- und Verkehrswert.

5. Bodenrichtwerte

Bodenrichtwerte sind Orientierungswerte zur Einschätzung des Marktwertes von Grundstücksflächen – ohne deren Gebäudewert. Bodenrichtwerte werden nach den Bestimmungen des Bundesbaugesetzes von den dazu bei den Städten und Kreisverwaltungen eingesetzten Gutachterausschüssen unter Verwendung der dafür vertraulich zugänglichen Kaufpreissammlungen ermittelt und veröffentlicht.

6. Markt- Preisspiegel

Die angegebenen Werte gelten für das jeweilige Gemeindegebiet und basieren auf den Marktberichten der jeweiligen Gutachterausschüsse.

Neben den Mietspiegelwerten für Wohnobjekte (mit Bezugsgröße 60-90 qm) werden Kaufpreise für Eigentumswohnungen, Reihenhäuser / Doppelhaushälften und Ein- Zwei-Familienhäuser angegeben.

7. Marktdaten

Marktdaten zeigen die Entwicklung folgender Werte einer Gemeinde seit 1975 bis 2001 an: Mieten für Eigentumswohnung, Neubau und Altbau, Baugrundstücke, Einzelhandel in der City, Einzelhandel in Nebenlage, Büro, Gewerbegrundstücke und Kaufpreise für Reihenhäuser. Die Daten liegen nur für bestimmte Städte in Ballungszentren vor.

8. Risiken

Unter dem Oberbegriff „Risiko“ werden die Werte spezieller Indikatoren ausgegeben, die wichtige Hinweise über die mit dem Objekt verbundenen Risiken liefern. Folgende Risiko-Indikatoren werden geliefert:

- Zahlungsrisiko drückt das statistische Ausfallrisiko für Kredite aus, die in dem Objekt wohnenden Konsumenten gewährt werden könnten.
- Umzugsvolumen beschreibt die absolute Menge der Zu- und Wegzüge insgesamt.
- Umzugssaldo drückt die durch die Umzüge bewirkte positive oder negative Veränderung des Bestandes aus.
- Fluktuation drückt die Relation von Umzugsvolumen zum Bestand aus.

9. Firmendaten

Die Angaben stützen sich auf eine Auswertung der Firmendatenbanken von Creditreform und weiteren Partnern. Soweit die Crefo-Firmenidentifikationsnummer ausgewiesen ist, kann eine zusätzliche Firmenabfrage abgerufen werden, die neben der Branche auch den ungefähren Umsatz des Vorjahres und die Mitarbeiterzahl angibt.

10. Branchen-Umfeld

Der Benutzer findet durch Vorgabe eines Ausschnittes um das Objekt herum und nach Auswahl einer Branche, ob und welche Firmen vorhanden sind.

11. Offline-Dienst

Dieser Bereich bietet dem Benutzer die Möglichkeit drei verschiedene Dienste bei einem Partner per eMail anzufordern.

- aktueller Grundbuchauszug für das Objekt (wenn die Berechtigung vorliegt)
- eine aktuelle Ermittlung zum gültigen Planungsrecht, sowie
- Ergebnisse einer speziellen Besichtigung des Objektes samt aktuellen Objekt-Fotos

5. Produkt-Daten

Die Daten werden in einem relationalen Datenbankmodell¹²⁹ verwaltet (SQL-Server 2000). Der Datenbestand des Datenbank-Servers wird in zwei Datenbanken aufgeteilt:

1. im_basis – Diese Datenbank enthält Grunddaten, dazu gehören Daten, die Informationen über den angemeldeten Benutzer enthalten (Aufbau siehe Abbildung 1):
 - im_user_daten Die Benutzer-Tabelle enthält Daten über die Benutzer.
 - im_firmen_daten Firma-Daten
 - im_firmen_kostenstellen Benutzer können verschiedenen Kostenstellen zugewiesen werden, das ist nur für die Abrechnung.
 - im_adresse Anschriften der Firmen und Benutzer (Außendienst)
 - im_kommunikation Zusätzliche Benutzer-Daten, hier werden z.B. Telefon-Nr, eMail-Adresse, Handy-Nr ... eingetragen.
 - im_grund_einstellungen enthält eine erweiterbare Liste von Daten, wie z.B.: IP-Adressen der Partner-Server, bestimmte Vorgaben zum Aufbau der Internetseiten
 - im_user_erste_seite ermöglicht die Unterscheidung zwischen „normalem“ Benutzer, Testbenutzer oder Administrator. Die verschiedenen Benutzer werden nach der Anmeldung auf unterschiedliche Seiten geleitet bzw. haben auf den Seiten unterschiedliche Berechtigungen.
 - im_status enthält verschiedene Berechtigungen (z.B. Testuser), die dem Benutzer zugeordnet werden.
 - im_grund_kosten enthält die Kosten „pro Click“ der einzelnen Dienste.
 - im_kosten_kopf Kopf-Tabelle des Billing- oder Abrechnungssystems. In dieser Tabelle wird pro angewähltem Objekt des Benutzers ein Tupel angelegt.
 - im_kosten_pos Positions-Tabelle des Billing-Systems. Wird für jeden „Click“ des Benutzers ein Datensatz gespeichert, um

¹²⁹Im **Relationalen Datenbankmodell** (entwickelt 1970 von E. F. Codd) werden die Daten in Tabellen (Relationen) gespeichert. Die Tabellen sind weitgehend unabhängig voneinander.

- später eine detaillierte Auflistung der Kosten erstellen zu können, wann und was ausgewählt wurde.
- im_hilfe_texte enthält alle Hilfetexte.
 - im_button_info enthält die Namen der Dienste und die Reihenfolge der zugehörigen Button in der Auswahl-Liste (Abb. 2).
 - im_kompetenz_gruppen enthält alle verschiedenen Kompetenz-Gruppen. Der Primärkey wird in der Tabelle im_user_daten als ID_kompetenzen hinterlegt.
 - im_kompetenz enthält für alle Kompetenz-Gruppen die Dienste, die angewählt werden dürfen.
2. im_objekt – Diese Datenbank enthält Tabellen, die primär die Informationen der Partner betreffen. Die Daten zweier Partner liegen sogar vollständig vor.
- marktdaten enthält den Datenbestand eines Partners. Die Daten geben Auskunft über die Kosten-Entwicklung von Miet- und Kaufobjekten von 1975 bis 2001.
 - haus, haus_kern, ... in 29 Tabellen ist der Datenbestand eines Partners enthalten. Die Tabellen beinhalten Textinformationen zu den 17,3 Millionen Objekten Deutschlands. Sie werden primär für die Dienste: Wohnumfeld, Risiken und Firmen benötigt.
 - im_teleinfo_staedte enthält die Städtenamen, für die es Straßenbilder gibt.
 - lage_texte enthält Beschreibungstexte für die Bilder eines Partners

Die Tabellen der Grunddaten-Datenbank (im_basis) enthalten sowohl Stamm- als auch Bewegungsdaten. Die Tabellen (im_kosten_kopf und im_kosten_pos), die das Abrechnungssystem verwalten, werden durch die Aufrufe des Benutzer erweitert und sind somit Bewegungsdaten. Die anderen Tabellen werden selten geändert, z.B. bei Preisänderungen, neue Tabellen werden höchstens nötig, wenn ein neuer Dienst eingebunden wird.

Die Tabellen der Objektdaten-Datenbank (im_objekt) werden nur dann geändert, wenn von einem Partner neue Datenbestände vorliegen. In diesem Falle werden dann komplette Tabellen ausgetauscht, möglicherweise auch neue Tabellen hinzugefügt.

6. Produkt-Leistungen

Die Verarbeitungsgeschwindigkeit und Stabilität des Software-Systems ist den üblichen Standards entsprechend. Da einige Dienste Bilddateien liefern, kann es zu Wartezeiten kommen, die bis zu 30 Sekunden dauern können, insbesondere wenn der Benutzer über eine analoge Telefonverbindung IMMOMENT aufgerufen hat.

Nach der Adressprüfung ist von den meisten Diensten bekannt, ob diese für das ausgewählte Objekt zur Verfügung stehen. Steht ein Dienst nicht zur Verfügung, ist der Button nicht anwählbar und das ist dem Benutzer durch einen grauen Button optisch kenntlich gemacht. Wenige Dienste können erst bei Anwahl geprüft werden. Somit erscheint, wenn dieser Dienst nicht zur Verfügung steht, ein entsprechender Hinweis auf der angewählten Seite. Darf der Benutzer durch seine Kompetenzstufe einen Dienst nicht anwählen, so wird dem Benutzer dieses durch einen grauen Button optisch kenntlich gemacht.

Alle angeforderten Dienste (Button-Clicks) werden protokolliert. Dabei wird bei nicht vorhandenen Informationen eine Fehlertabelle fortgeschrieben. Die Protokoll-Tabelle, die die erfolgreichen Dienste-Abfragen enthält, dient der Rechnungserstellung und gehört zum Abrechnungssystem.

7. Benutzeroberfläche

Für die Online Bedienung ist eine graphische Benutzeroberfläche obligatorisch. Diese kann durch die Wahl des Browser vom Benutzer mitbestimmt werden.

Vorausgesetzt, der Benutzer ist Windows-orientiertes Arbeiten (Hyperlinks, Buttonclick, Umgang mit der Maus) gewohnt, so ist im Grunde ein intuitives Arbeiten möglich.

Der Bildschirmaufbau ist so gewählt, dass auf jeder Internet-Seite die ausgewählte Objektadresse zu sehen ist, damit der Benutzer jederzeit erkennt, zu welchem Objekt die angezeigten Daten gehören. Auf jeder Seite stehen Hilfetexte zur Verfügung, die sowohl die Benutzung als auch die angezeigten Daten erläutern.

Ein Prototyp zeigte bereits im Vorfeld, dass sowohl bestimmte optische als auch funktionale Änderungen vorgenommen werden mussten.

8. Qualitätsbestimmung

Zu diesem Punkt kann keine eindeutige Aussage getroffen werden, da die Haupteinschätzung der Qualität durch die Partner-Daten von IMMOMENT bestimmt wird. Die Partner sind derzeit noch im Aufbau ihrer Daten, so können z.B. Bildinformationen für ein angefragtes Objekt primär nur in zwei Bundesländer (NRW und Niedersachsen) und in Ballungsräumen abgefragt werden.

Die Qualitätsbestimmungen zum funktionalen Ablauf können für die folgenden Punkte nach dem üblichen Schulnotenverfahren mit „gut“ bewertet werden:

- Produktqualität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit

Dabei hängt der Punkt Zuverlässigkeit aber auch von dem Provider ab, bei dem IMMOMENT gehostet wird.

9. Globale Testfälle

In einem ersten Schritt wurde ein Prototyp erstellt. Daraus ergaben sich Änderungsanforderungen, die in einer ersten Testversion online mehreren Kunden zur Verfügung gestellt wurden.

Parallel zur Testversion wurde eine Demoversion erstellt. Diese diente hauptsächlich dazu, dass ein Interessent sich an einigen Testadressen den Leistungsumfang veranschaulichen konnte und die dialogorientierte Benutzungsoberfläche getestet wurde. Diese Demo kann vom Web-Server heruntergeladen werden und ist dann offline in allen üblichen Browsern auf dem Client-Computer lauffähig.

So konnte über Wochen die Testversion optimiert werden, dabei erweiterte sich auch der Umfang der Anwendung durch weitere Partner.

10. Entwicklungsumgebung

Die Web-Applikation IMMOMENT wurde komplett mit dem neuen Microsoft Produkt .Net unter C# als Web-Applikation erstellt. C# ist eine C++ und Java ähnliche objektorientierte Programmiersprache, die mit der Entwicklungsumgebung Visual Studio 7.0 unterstützt, das schnelle Erstellung von Oberflächen und der kompletten Anwendung ermöglichte.

IMMOMENT stellt als „Informationssammler“ von Partner-Web-Servern eine besondere Art der Programmarchitektur dar, so dass sich sehr früh herausstellte, dass „alte“ Anwendungen, besser gesagt bereits vorhandene globale Strukturen und komplexe mathematische Berechnungsfunktionen, nicht genutzt werden konnten. Aus diesem Grunde ergab sich die Möglichkeit ein neues Produkt zur Erstellung von IMMOMENT einzusetzen, da keine vorhandenen „Pfade“ wie Schnittstellen oder Architekturen berücksichtigt werden mussten.

Ein weiterer Grund, der dafür sprach einen neuen Weg einzuschlagen, war, dass festgestellt wurde, dass die Web-Applikation mit den derzeitigen Erstellungsmethoden (asp-Seiten mit XSL und Html Seiten – komplexere Logik durch com-Programme per C++) einen zu großen Aufwand bedeutet hätte, so dass eine erste lauffähige Version von zwei Programmierern in etwas mehr als zwei Monaten nicht hätte erstellt werden können, da ein fester Termin für eine erste Version vorgegeben war.

Des Weiteren hatten Schulungen gezeigt, dass gerade die Erstellung der Oberflächen mit Visual Studio sehr schnell möglich ist. So wurde die Entscheidung gefällt, IMMOMENT mit einer komplett neuen Umgebung und Programmiersprache zu erstellen. IMMOMENT ist die erste Anwendung, die von der Firma IMMO-DATA AG in .Net und C# erstellt wurde.

11. Ergänzungen

In den vorherigen Punkten ist alles ausführlich erwähnt worden, somit gibt es keine Ergänzungen.

Anhang B – Literaturverzeichnis

Nr.	Autor(en) (Erscheinungsjahr) : Titel	Fußnote
1	Balzert, H. (1996): Lehrbuch der Software-Technik: Software-Entwicklung. Heidelberg.	1, 7, 25
2	Rechenberg, Pomberger (2002, 3. Auflage) : Informatik-Handbuch	14, 26, 27
3	Prof. Dr. Wilmes, T. (1999) : Softwareentwicklung – Skripte zum Verbundstudium Wirtschaftsinformatik VBWI (000535 1. Lerneinheit)	26, 28 – 32
4	Prof. Dr. Faeskorn-Woyke, H. (1999) : Datenbanken und Informationssystem – Skripte zum VBWI (000596 – 1. LE + 000603 – 4. LE)	20, 42, 44, 45, 50, 51
5	Prof. Dr. rer. Oec. Bernd Blümel (1999) : Software-Engineering - Skripte zum VBWI (000457 – 4. LE)	107
6	Prof. Dr. Schäfer-Richter, G (1997): Informatik B – Skripte zum VBWI (000108 – 1. LE)	90 + 91
7	Prof. Dr. Görner, U. (1999) : Informationsmanagement - Skripte zum VBWI (000554 – 1. LE)	5
	Es folgen die Internetseiten – die zitierten Inhalte sind alle Ende Mai 03 geprüft worden und waren zu dem Zeitpunkt vorhanden.	
8	Hansen, H. R. (1997 - 7.Auflage) : "Wirtschaftsinformatik I", von http://gd.tuwien.ac.at/study/hrh-glossar/ Entwicklung und Implementierung einer Internet-Version des Glossars zum Buch der TU Wien	11, 13, 19, 21, 40, 41, 86, 109
9	Kosch, A. (2002) : Crashkurs .NET für Delphianer – http://software-support.biz/itr/buecher/download/probekapitel_crash/CrashKursK1.htm	36 + 37
10	Internetseite der Sparkasse Eichsfeld: www.kreissparkasse-eichsfeld.de/framescout.html?061P00BhM.html	6 + 9
11	cojobo.bonn.de/projekte/eu.htm	121
12	www.oreilly.de/xml/xml_faq_a1.html	14
13	www.microsoft.com/germany/ms/services/support/premier/developer/default.htm	38
14	www.commando.de/glossar/gloss-w.htm	15
15	www.versata-deutschland.de/products/webservice.html	17
16	wwwswt.informatik.uni-rostock.de/deutsch/Mitarbeiter/michael/lehre/Usab_WS2000/Vortrag11/vortrag.htm	22
17	www.cavallinoqualitycamping.com/deu/manifesto_iso.htm	53
18	www.bsi.de/fachthem/egov/download/4_IntAuf.pdf	65
19	www.www-kurs.de/cookies.htm	69
20	www.dclp-faq.de/q/q-php-was.html	76
21	www.asphelper.de/ASP Kurs/A100100.asp	77
22	www.general.co.at/netsecurity.nsf/0/AFBA5F8DA08404A9C12567DA00399DC4?OpenDocument	79, 80, 82
23	www.muelheim-ruhr.de/f93b50fc873425da2b2e7fd301c7db72.html	81
24	www.geographie.uni-mannheim.de/swiarzny/MLV/gk_koord.htm	100
25	www.lverma.nrw.de/vermessung/raumbezug/KO_MERKB.htm	100
26	www.idg.de/de/fln/allgemein/datenschutz_und_sicherheit/datenschutz_und_sicherheit.htm	104
27	www.wise-ftp.de/faq_Eintrag8.htm	108
28	www.commando.de/glossar/gloss-u.htm	115

Anhang C – Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Hagen, den

Michael Drzasga