



Hochschule Neubrandenburg
University of Applied Sciences

Fachbereich Landschaftsarchitektur, Geoinformatik, Geodäsie,

Bauingenieurwesen

Studiengang Geoinformatik

Entwicklung eines Hochwasserinformationssystems

Am Beispiel der Aland-Mündung

Bachelorarbeit

zur Erlangung des akademischen Grades „**Bachelor of Engineering**“ (B.Eng.)

vorgelegt von:

Fabian Schulte

URN: urn:nbn:de:gbv:519-thesis2011-0528-2

Erstprüfer: Prof. Dr.-Ing. Andreas Wehrenpfennig

Zweitprüfer: Dipl.-Ing. oec. Hans-Jörg Steingraf

Neuruppin, September 2011

Kurzzusammenfassung

Im Rahmen dieser Bachelorarbeit wurde ein Hochwasserinformationssystem entworfen und implementiert. In dem Projekt wurde der Flussabschnitt des Alands im Landkreis Stendal (Sachsen-Anhalt) zwischen Seehausen und der Mündung in die Elbe mit wasserwirtschaftlichen Bauwerken, wie Deichen, Poldern, Wehren und Pegeln dargestellt. An bestimmten Bauwerken sollen mithilfe einer Datenbank-Anbindung die aktuellen Bauwerksdaten, wie Pegelstände oder Wehrstatus dargestellt werden. Eine weitere Funktion ist die Visualisierung von Wasserständen in den Polderflächen des Polders Garbe und Wrechow. Dazu gibt der Nutzer entweder einen Höhenwert für den Pegel des Polders in das Web-Interface ein oder der Pegelstand wird aus der Datenbank ausgelesen. Daraufhin wird angezeigt, welche Flächen in welcher Tiefe unter Wasser stehen. Die Visualisierung erfolgte mithilfe von Google Maps API innerhalb des Web-Projekts.

Abstract

Within the scope of this bachelor thesis a flood-information-system was designed and implemented. Project territory is the reach of river Aland in the administrative district of Stendal, part of the federal state of Saxony-Anhalt between Seehausen and the water mouth to the river Elbe. In this territory all water management buildings like levees, polder areas, weirs and gauges are presented. At some buildings the level of the gauges or the state of the weir will be displayed with the help of a database connection. Another feature is the visualization of water levels in the polder areas of polder Garbe and polder Wrechow. To do this the user enters the level of gauge manually or the level is read out of the database. This shows which and how depth areas are flooded. The visualization is realized with Google Maps API.

INHALTSVERZEICHNIS

1	Einführung in die Problemstellung.....	5
1.1	Hinführung zum Thema.....	5
1.2	Problemstellung	5
1.3	Ziel der Arbeit.....	6
2	Grundlagen	7
2.1	Vorstellung des Projektgebietes.....	7
2.2	Die LHW Hochwasservorhersagezentrale (HVZ) als Beispiel eines HIS... 8	8
2.3	Begriffsdefinitionen.....	10
2.3.1	Hochwasser.....	10
2.3.2	Wehr.....	10
2.3.3	Hochwasserschutzdeich	11
2.3.4	Retentionsraum	11
2.3.5	Siel.....	11
2.3.6	Pegel	11
2.4	Online-Kartenanwendungen und Google Maps	11
2.5	Google Maps JavaScript API Version 3	13
2.6	Keyhole Markup Language (KML).....	13
3	Verwendete Software und Technologien	14
3.1	Verwendete Software.....	14
3.1.1	Geoinformationssysteme und ArcGIS 9.3.....	14
3.1.2	Die NetBeans IDE.....	16
3.1.3	XAMPP	16
3.2	Technologien.....	17
3.2.1	PHP	17
3.2.2	JavaScript.....	17
3.2.3	MySQL.....	18
4	Planung.....	19
4.1	Geforderte Funktionalitäten des Hochwasserinformationssystems.....	19
4.2	Datenbankentwurf.....	21
5	Transformation von Rasterdaten nach KML.....	27
5.1	Verarbeitung der Rasterdaten	28
5.2	Transformationen zwischen den Koordinatensystemen	29
5.3	Ausgabe der Daten in KML - Dateien.....	30
6	Entwicklung eines Prototypen	32
6.1	Die MySQL-Datenbank	32

6.2	Architektur und Struktur des Web-Interfaces.....	33
6.3	Interaktion zwischen PHP-Seiten	35
6.4	Entwurf eines Layouts	36
6.5	Administrator-Bereich.....	37
6.5.1	Anlegen eines Bauwerks	38
6.5.2	Einlesen / Eingeben von Daten zu einem Bauwerk	39
6.5.2.1	Pegelwerte manuell eingeben	39
6.5.2.2	Pegelwerte aus Excel einlesen	40
6.5.2.3	Aktion an einem Wehr durchführen.....	40
6.5.3	Datenbank bereinigen.....	41
6.6	Die Visualisierung der Bauwerke	41
6.7	Die Visualisierung von Überflutungsflächen	43
6.7.1	Manuelle Pegelstandseingabe.....	44
6.7.2	Visualisierung durch Datenbank-Pegelwerte	45
7	Zusammenfassende Darstellung und Ausblick	46
7.1	Zusammenfassung der Ergebnisse.....	46
7.2	Ausblick.....	46
	Literaturverzeichnis	48
	Abbildungsverzeichnis.....	50
	Tabellenverzeichnis.....	52
	Anhang.....	53

1 EINFÜHRUNG IN DIE PROBLEMSTELLUNG

1.1 Hinführung zum Thema

„Hochwasser sind Bestandteile des natürlichen Wasserkreislaufes und daher nicht zu vermeiden.“ (Patt, 2001, S. 1) Diese Aussage von Heinz Patt stellt die übergeordnete Problematik dar, woraus die Notwendigkeit resultiert, sich vor den Hochwasserereignissen zu schützen, bzw. dessen Auswirkungen so gut wie möglich einzudämmen. Der Landesbetrieb für Hochwasserschutz und Wasserwirtschaft Sachsen-Anhalt (LHW) ist mit den Aufgaben des staatlichen Wasserbaus, der Gewässerunterhaltung, dem Hochwasserschutz, dem Gewässerkundlichen Landesdienst, Wasserlaboren und dem hydrologischen Dienst betraut.

1.2 Problemstellung

„Bis zum Bau des linken Alanddeiches staute die Elbe bei höheren Wasserständen (heute entsprechend etwa ab 500 cm am Pegel Wittenberge) in den Aland zurück und strömte über das linke Alandufer in die Schau- und Zehrengabenniederung. Über die Seegeniederung gelangte das Wasser wieder in den Elbelauf zurück. Diese Wasserstände treten etwa alle 1-2 Jahre auf.“ (Priebe, 2009, S. 1)

Der Aland, ein westlicher Nebenfluss der Elbe, wurde im 19. Jahrhundert bis Seehausen beidseitig eingedeicht, dass er in die Alandmündung einströmendes Elbehochwasser bis 630 cm am Pegel Wittenberge mit Sicherheit kehren, zurückhalten, konnte. Eine Studie aus dem vergangenen Jahrhundert zeigt, dass jede Belastung des linken Alanddeiches über einen Wasserstand von 20,30m ü. NN (630 cm am Pegel Wittenberge) eine akute Gefährdung darstellt. „Im Falle des Eintritts eines Elbehochwassers mit einem Scheitelwert > 680 cm am Pegel Wittenberge und vollem Rückstau in den Aland ist die Möglichkeit des Versagens der Alanddeiche außerordentlich wahrscheinlich.“ (Priebe, 2009, S. 1) Seit 1991 ist das Alandabschlussbauwerk funktionstüchtig und sorgt dafür, dass das Einströmen der Elbe in den Alandschlauch verhindert wird. Der LHW, als die für den Hochwasserschutz an Elbe und Aland zuständige Einrichtung hat das Ziel, den Schutz der Altmärkischen Wische vor Hochwasser der Elbe und des Alands zu gewährleisten. Dadurch sollen die Niederungen des Aland, des Schau- und Zehrengabens und der Seege bis zu einem Bemessungswasserstand der Elbe

am Pegel Wittenberge vor Deichbrüchen und unkontrolliertem Überströmen der Deiche geschützt werden.

1.3 Ziel der Arbeit

Im Rahmen der Bachelorarbeit soll ein Hochwasserinformationssystem (HIS) entworfen und implementiert werden. Dies wird durch ein Web-Interface realisiert, in das der Flussbereich des Alands von Seehausen bis zur Mündung in die Elbe visualisiert wird. Eine Funktion des Hochwasserinformationssystems ist die Darstellung von Wasserständen in den Überschwemmungsflächen Polder Garbe und Wrechow. Dabei soll der Nutzer die Möglichkeit haben Pegelstände einzugeben und sich die überfluteten Flächen als Simulation anzeigen zu lassen. Eine weitere Funktion stellt das Anzeigen von Pegelständen an verschiedenen wasserwirtschaftlichen Bauwerken dar. Ein geschützter Bereich gibt dem Administrator des Hochwasserinformationssystems die Möglichkeit neue Bauwerke, wie Pegel oder Wehre, einzurichten, sowie dessen Pegelwerte mithilfe eines Formulars einzulesen bzw. einzugeben oder Aktionen am Wehr (öffnen/schließen) durchzuführen.

Der Nutzer kann sich diese Pegelmesswerte und den Wehrstatus anzeigen lassen, aber auch eine Darstellung der Überflutungsflächen in den Poldern anhand jener Pegelwerte ist möglich. Außerdem soll es möglich sein, die angezeigten Pegelwerte oder Aktionen am Wehr in einem PDF-Format tabellarisch aufgelistet zu exportieren.

Die Visualisierung soll mithilfe der Google Maps API umgesetzt werden. Die Bauwerksdaten werden in einer MySQL-Datenbank gespeichert und langfristig gesichert.

2 GRUNDLAGEN

2.1 Vorstellung des Projektgebietes

Das Projektgebiet ist im Norden Sachsen-Anhalts, im Landkreis Stendal lokalisiert. Dieser grenzt im Osten an die Brandenburgischen Landkreise Ostprignitz-Ruppin und Havelland, im Norden an den Landkreis Prignitz. Im Nordwesten an den Niedersächsischen Landkreis Lüchow-Dannenberg, im Westen an den Altmarkkreis Salzwedel sowie im Süden an die Landkreise Jerichower Land und Börde. Im Rahmen der Bachelorarbeit soll der Flussbereich des Alands zwischen Seehausen (Altmark) und der Mündung in die Elbe (nahe Schnackenburg) betrachtet werden (siehe Abb. 2-1).

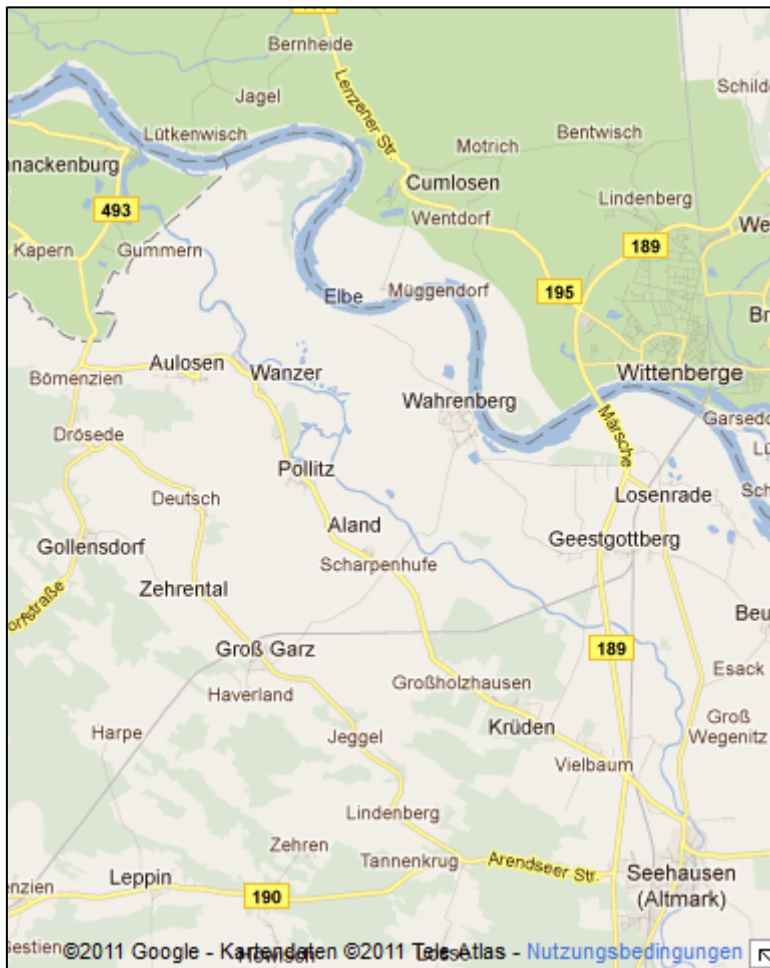


Abbildung 2-1 Projektgebiet (Google Maps, 2011)

2.2 Die LHW Hochwasservorhersagezentrale (HVZ) als Beispiel eines HIS

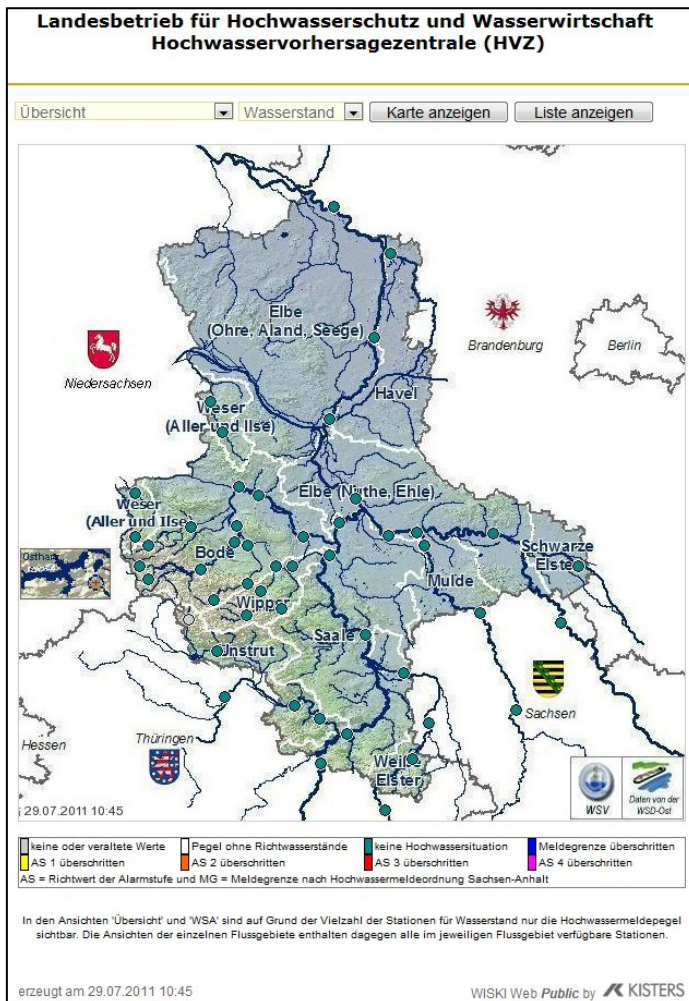



Abbildung 2-2 Kartenansicht Hochwasservorhersagezentrale (Landesbetrieb für Hochwasserschutz und Wasserwirtschaft, 2011)

Der Landesbetrieb für Hochwasserschutz und Wasserwirtschaft (LHW) unterhält bereits eine Hochwasservorhersagezentrale (HVZ). In der obigen Übersicht (siehe Abb. 2-2) finden sich alle Hochwassermeldepegel im Land Sachsen-Anhalt wieder.

Es besteht auch die Möglichkeit, in der Karte oder über die Dropdown-Liste auf einzelne Teilbereiche zu klicken, z.B. auf den Bereich „Elbe (Ohre, Aland, Seege)“, und sich diesen in der Listenansicht anzeigen zu lassen:

**Landesbetrieb für Hochwasserschutz und Wasserwirtschaft
Hochwasservorhersagezentrale (HVZ)**


SACHSEN-ANHALT

Elbe (Ohre, Aland, Seege) | Wasserstand | Karte anzeigen | Liste anzeigen

Gewässer	Station	Nummer	Zeitpunkt	Wasserstand	HW-Situation	Tendenz	-2h	-4h	-6h	-12h	-24h	MG	AS1	AS2	AS3	AS4	Abfluss	MQ
Elbe	Maadeburg	502180	29.07.2011 08:00	208 cm		↘	210	212	214	221	233							557
Elbe	Niegripp	502240	29.07.2011 08:00	412 cm		↓	418	421	424	432	448	640	740	820	900			
Elbe	Werben-Neu Werben EP	502470	29.07.2011 08:00	389 cm		↘	391	393	394	398	401							
Elbe	Havelberg-Schl. EP	502475	29.07.2011 08:00	354 cm			353	358	361	366	367							
Elbe	Quitzebel UP	580830	29.07.2011 08:00	332 cm		→	332	333	333	334	332							
Elbe	Tangermünde	502350	29.07.2011 08:00	340 cm		↘	341	343	346	353	363	500	600	660	700			570
Elbe	Wittenberge	503050	29.07.2011 08:00	348 cm		→	349	349	349	351	348	500	600	630	670			677
Elbe	Lenzen	503120	29.07.2011 05:00	345 cm		→	344	344	344	339	331							
Salzwedeler Dumme	Salzwedel-Dumme	597208	29.07.2011 05:30	13 cm			14	15	16	19	11							

keine oder veraltete Werte
 Pegel ohne Richtwasserstände AS 2 überschritten
 keine Hochwassersituation AS 3 überschritten
 Meldegrenze überschritten AS 4 überschritten
 AS = Richtwert der Alarmstufe und MG = Meldegrenze nach Hochwassermeldeordnung Sachsen-Anhalt

Die Tendenz wird für jeden Pegel aufgrund seines typischen Anstiegs- und Abfallverhaltens anhand von Berechnungsrastern ermittelt und ist damit nur eine automatisch berechnete Orientierungsgröße.


erzeugt am 29.07.2011 10:48 WISKI Web Public by 

Abbildung 2-3 Listenansicht Hochwasservorhersagezentrale (Landesbetrieb für Hochwasserschutz und Wasserwirtschaft, 2011)

In diesem Ausschnitt (siehe Abb. 2-3) befindet die Listenansicht aller Pegel im Bereich „Elbe (Ohre, Aland, Seege)“. Dieser ist mit einer Nummer genau definiert und gibt den aktuellen Wasserstand mit dazugehörigem Zeitpunkt an. Außerdem wurde eine Legende definiert, die die aktuelle Hochwassersituation beschreibt; eine Spalte zeigt die Tendenz des Pegels an. Die weiteren Spalten zeigen vorherige Pegelwerte in definierten Zeitabständen (Spalten „-2h“, „-4h“...), die Meldegrenze, die Werte die für eine bestimmte Alarmstufe notwendig sind (Spalten „AS1“ bis „AS4“), den Abfluss, sowie den mittleren Abfluss (Spalte „MQ“), an.

Ein Klick auf einen beliebigen Pegel zeigt diesen im Detail, mit unterschiedlichen Parametern, wie z.B. die Messstellen-Nr., Stationsname, Gewässer, Einzugsgebiet, Pegelnullpunkt, Lage am Gewässer, Zuständigkeit und Bundesland. Eine Grafik in Form einer Ganglinie, bzw. die gemessenen Pegelwerte in tabellarischer Form zeigen den Verlauf des Pegels der letzten sieben Tage an.

Vorteil der Hochwasservorhersagezentrale ist die hohe Aktualität durch die interne Verarbeitung und Ausgabe in WISKI Web. Nachteilig – nicht alle Gewässer von denen auch Pegeldata existieren (z.B. Aland) sind in die Hochwasservorhersagezentrale eingebunden.

2.3 Begriffsdefinitionen

2.3.1 Hochwasser

Als Hochwasser wird „die zeitlich begrenzte Überschwemmung von normalerweise nicht mit Wasser bedecktem Land durch oberirdische Gewässer oder durch in Küstengebiete eindringendes Meerwasser“ (Wasserhaushaltsgesetz - WHG, 2009) bezeichnet.

2.3.2 Wehr

„Wehre sind wasserbauliche Stauanlagen. (...) Stauanlagen sind Wasserbauwerke, mit deren Hilfe Bäche und Flüsse aufgestaut werden.“ (Strobl & Zunic, 2006, S. 125)

Es wird zwischen festen und beweglichen Wehren unterschieden. Feste Wehre haben keine mechanischen Einrichtungen, sie sind deshalb betriebssicher und billiger. An bewegliche Wehren sind Regulierorgane, so genannte Schützen, angebracht, durch die eine Regelung des Wasserstands ermöglicht wird.

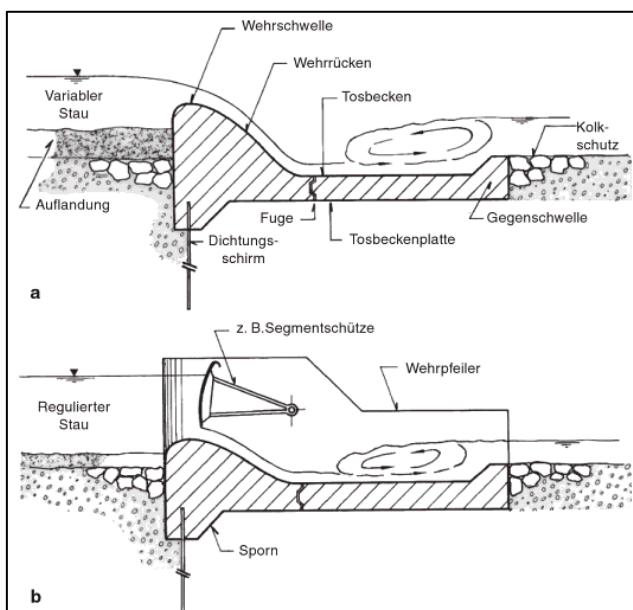


Abbildung 2-4 Querschnitt durch a ein festes Wehr und b ein bewegliches Wehr (Patt & Gonsowski, 2011, S. 122)

2.3.3 Hochwasserschutzdeich

„Deiche dienen zum Schutz vor Hochwassern und – an der Küste – vor Sturmfluten. Im Gegensatz zu Dämmen, die ein Gewässer dauerhaft aufstauen, sind Deiche nur während eines Hochwassers zeitweise eingestaut. (...) Deiche können als homogene Schuttkörper aus Erdbaustoffen erstellt werden. Besitzen sie keine spezielle Dichtung, werden Deiche beim Einstau sehr langsam durchströmt.“
(Strobl & Zunic, 2006, S. 430)

2.3.4 Retentionsraum

Als Retentionsraum wird der unbebaute, meist flache Raum bezeichnet, der bei Hochwasser überflutet werden kann, dazu gehören auch Polder. Er trägt somit zur Senkung des Wasserspiegels des zu entlastenden Gewässers bei. Diese werden gezielt als Überschwemmungsgebiete festgelegt. In hochwasserfreien Zeiten können diese Flächen als Grünflächen und durch forst- und landwirtschaftliche Nutzung als Kulturland dienen.

2.3.5 Siel

Als Siel wird der verschließbare Durchlass für Gewässer in einem Deich bezeichnet. Siele können sowohl Auslässe als auch Einlässe sein. Durch die Verschlussvorrichtung kann ein Rückstau aus dem Hauptgewässer in das Nebengewässer verhindert, sowie Abfluss und Wasserspiegel im Nebengewässer geregelt werden, z.B. zur Entwässerung eingedeichter Polder.

2.3.6 Pegel

Ein Pegel ist ein Gerät zur Bestimmung des Wasserstandes in unterschiedlichen Gewässern. Allgemein bezeichnet das Wort Pegel auch die Messstelle selbst.

2.4 Online-Kartenanwendungen und Google Maps

Die Bedeutung von Geodiensten hat in den vergangenen Jahren enorm zugenommen. Geodienste bilden das Bindeglied zu den Geodaten. Für den Zugriff auf die Geodienste können Webapplikationen oder lokal installierte GIS-Programme verwendet werden.

In der heutigen Zeit sind Online-Kartenanwendungen allgegenwärtig. Sie werden genutzt um den Ort von Veranstaltungen oder ähnlichem zu sehen, um Positionen von Adressen zu suchen, zum Navigieren zu Fuß oder mit Fahrzeugen. Auch in sozialen Netzwerken, wie Facebook oder Google+, ist es möglich die aktuelle Position per Smartphone zu übermitteln um die aktuelle Position mitzuteilen. Es gibt viele Online-Kartendienste, wie Yahoo! Maps, Bing Maps von Microsoft, aber am meisten wird der im Februar 2005 gegründete Dienst Google Maps genutzt. Maßgebenden Anteil an diesem Erfolg hatte die einfache Erstellung von sogenannten „Mashups“, eine „Kombination verschiedener Webservices über eine offene Programmierschnittstelle (API).“ (DATACOM Buchverlag GmbH) Jeder Internetnutzer wird bereits mit diesen Mashups in Kontakt gekommen sein, in dem Zusatzinformationen über Google Maps beispielsweise zum aktuellen Kinoprogramm, Immobilienangeboten oder Informationen über Sehenswürdigkeiten, Restaurants und Kulturveranstaltungen abgerufen wurden.

„Google Maps war der erste Kartendienst, der Satellitenbilder für alle Regionen der Erde zur Verfügung stellte und als reine Internetanwendung ohne eine spezielle Software verwendbar war.“ (Gunkel, 2010, S. 4) Im April 2006 standen dann hochauflösende Satellitenbilder von Deutschland bereit, allerdings vorerst in der BETA-Phase, die sich noch bis September 2007 hinzog.

Die Google Maps API ist laut (ProgrammableWeb) die beliebteste aller APIs. Mit 41% stellt Google Maps mit Abstand die meisten Mashups. Im Vergleich dazu, Microsofts Pendant Bing Maps mit 3%, zeigt die Dominanz Googles deutlich.

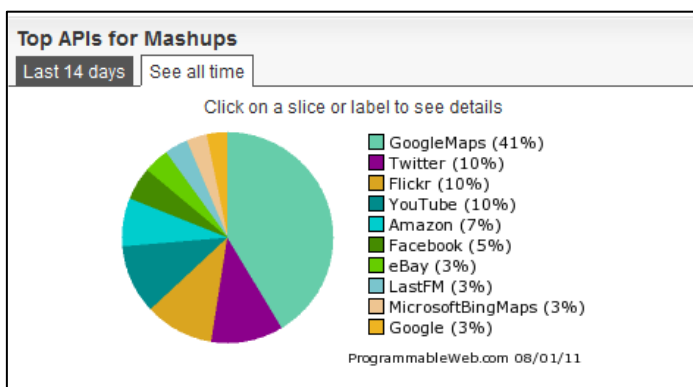


Abbildung 2-5 Die beliebtesten APIs als Mashups (ProgrammableWeb, 2010)

2.5 Google Maps JavaScript API Version 3

Bei den Google Maps APIs (API=*Application Programming Interface*) „handelt es sich um eine Reihe von Services, die es (...) ermöglichen, Karten, Geocodierung und andere Inhalte von Google in (...) Webseiten oder Anwendungen zu übernehmen.“¹ Google Maps besitzt eine Reihe von APIs mit dessen Hilfe es in die eigene Webseite eingebettet, sowie mit eigenen Daten versehen werden kann. Zur Google Maps API-Familie gehören die JavaScript-API, die API für Flash, die Google Earth API, die Static Maps API, Web-Dienstleistungen und die Google Maps Data-API.

Im Rahmen der Bachelorarbeit wird „Google Maps JavaScript API V3“ zur Einbindung in das Web-Interface verwendet. Zuvor war es für jede Webseite, auf der man Google Maps einbinden wollte, notwendig, sich einen API-Schlüssel für diese Seite von Google erstellen zu lassen. Dazu bedürfte es, die Webseite, auf der die API zum Einsatz kommen sollte, zu registrieren. Dies entfällt seit der Version 3. Google Maps basiert auf der AJAX-Technologie (**A**synchronous **J**avascript and **X**ML), die das dynamische Laden der Karte ermöglicht, ohne die gesamte Webseite neu laden zu müssen.

2.6 Keyhole Markup Language (KML)

Um Geodaten in Google Earth und Google Maps darzustellen, wird KML verwendet. KML basiert auf dem XML-Standard und verwendet eine Tag-basierte Struktur, mit verschachtelten Elementen und Attributen (siehe Anhang Abbildung A-1). Als Vorlage dient eine Google KML Referenz, in der alle Tags definiert wurden. In dieser stehen auch optionale Tags. Folgende Features können mit KML dargestellt werden:

- Orstmarken
- Beschreibungen
- GroundOverlay
- Linien
- Polygone

¹ (Google Maps-/Google Earth-APIs – Nutzungsbedingungen, 2011)

3 VERWENDETE SOFTWARE UND TECHNOLOGIEN

3.1 Verwendete Software

3.1.1 Geoinformationssysteme und ArcGIS 9.3

“Für eine Lange Zeit haben Menschen die Welt mit Modellen anhand von Karten und Globen studiert. In den letzten dreißig Jahren wurde es möglich diese Modelle auf Computern zu bringen (...). Diese Modelle, einhergehend mit Tools für dessen Analyse, führten zu einem Geoinformationssystem (GIS).“ (Ormsby, 2004, S. 1)

Das GIS ermöglicht die entsprechenden Daten zu organisieren, zu analysieren, zu modellieren; aber auch zu präsentieren, sei es grafisch oder tabellarisch. Der größte Vorteil ist das Zuweisen von Attributen zu geografischen Objekten, sodass zu einzelnen Punkten viele Informationen gespeichert; und aus dessen Vielzahl Informationen erlangt werden können.

Die wohl bedeutendste Softwarelösung auf dem Gebiet der Geographischen Informationssysteme ist die Produktpalette ArcGIS Desktop der Kalifornischen Firma Environmental Systems Research Institute (ESRI), mit Sitz in Redlands.

Bei der ArcGIS Desktop-Reihe gibt es verschiedene Produkte, die eine zunehmende Funktionalität aufweisen. Beginnend beim ArcReader, mit dem lassen sich veröffentlichte Dokumente aus ArcMap jedoch nur ansehen und drucken (vergleichbar mit dem Adobe Reader für PDF-Dokumente). Der ArcReader ist frei verfügbar und kann kostenlos auf der ESRI-Webseite heruntergeladen werden.

Die nächste Produktstufe ist ArcView. In ArcView können raumbezogene Daten eingelesen, verwaltet, analysiert und dargestellt bzw. ausgegeben werden. Anhand der raumbezogenen Daten können diese mit ArcView so analysiert werden, dass Daten gewonnen werden, die aus einfachen Tabellenkalkulationen oder Datenbanken so nicht hervorgehen würden²

Mit dem ArcEditor nehmen die Funktionen für die „Definition, Erfassung, Bearbeitung und Qualitätssicherung von Geodaten“ (ESRI Deutschland, 2011) noch einmal zu. Darauf soll nun nicht näher eingegangen werden.

² Vgl. (ESRI Deutschland, 2011)

ArcInfo ist das komplette Paket, das die ArcEditor-Funktionalitäten um alle Funktionen für die räumliche Analyse erweitert.

Alle Pakete besitzen jedoch eine gemeinsame Schnittstelle (z.B. ArcMap und ArcCatalog), die einen unkomplizierten Austausch zwischen den verschiedenen Produktlinien ermöglicht. In ArcMap werden die raumbezogenen Daten visualisiert, editiert und analysiert.

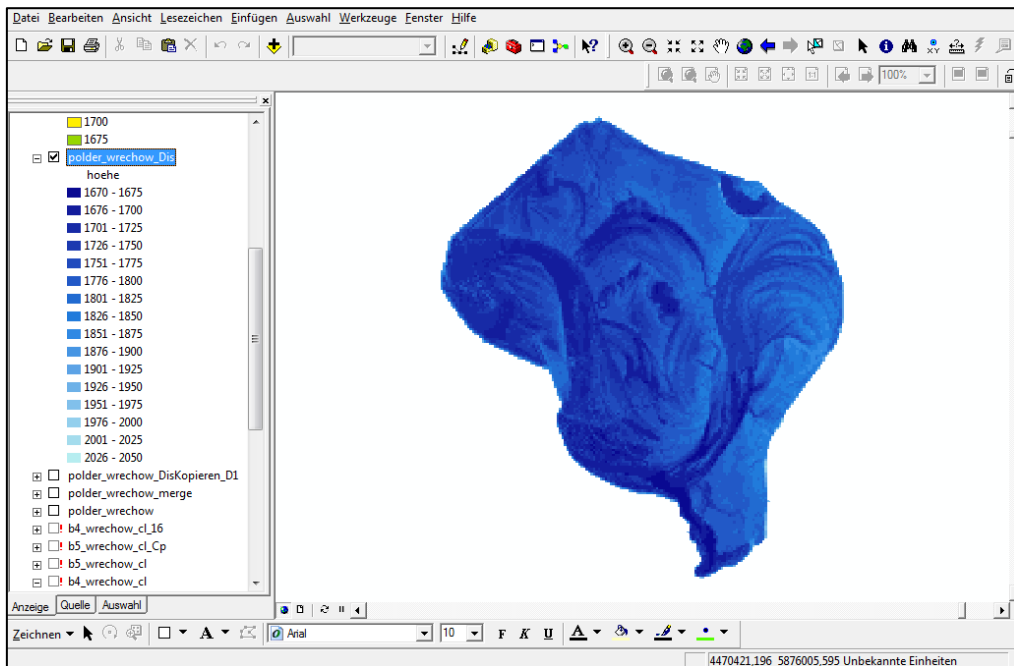


Abbildung 3-1 Ausschnitt aus ArcMap

Der ArcCatalog dient dem Organisieren der Daten (vergleichbar mit dem Windows Explorer).

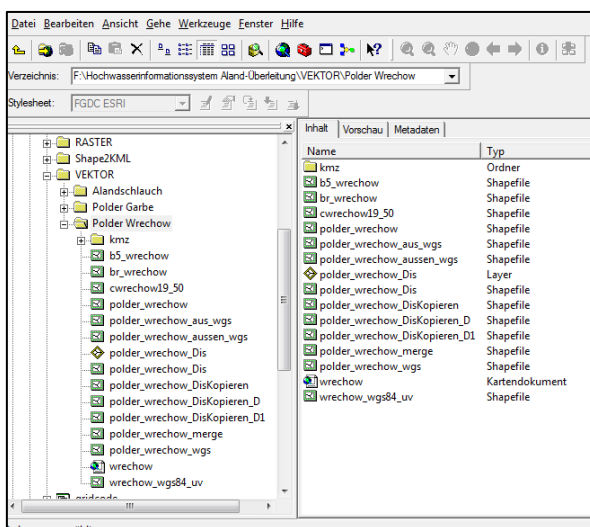


Abbildung 3-2 Ausschnitt aus ArcCatalog

3.1.2 Die NetBeans IDE

NetBeans ist ein Open Source Projekt, das im Jahre 2000 von Sun Microsystems gegründet wurde. 2010 wurde Sun Microsystems von Oracle übernommen und Oracle ist nun Hauptsponsor des Projekts.

Die NetBeans IDE dient Programmierern als Entwicklungsumgebung, „um Programme zu schreiben, zu kompilieren, zu testen, zu debuggen, zu profilieren, und bereitzustellen. Sie ist in Java geschrieben - unterstützt aber jede Programmiersprache. Es existiert eine große Zahl von Modulen, um die NetBeans IDE zu erweitern. Die NetBeans IDE ist ein kostenloses Produkt ohne Nutzungsbeschränkungen.“ (NetBeans, 2011)

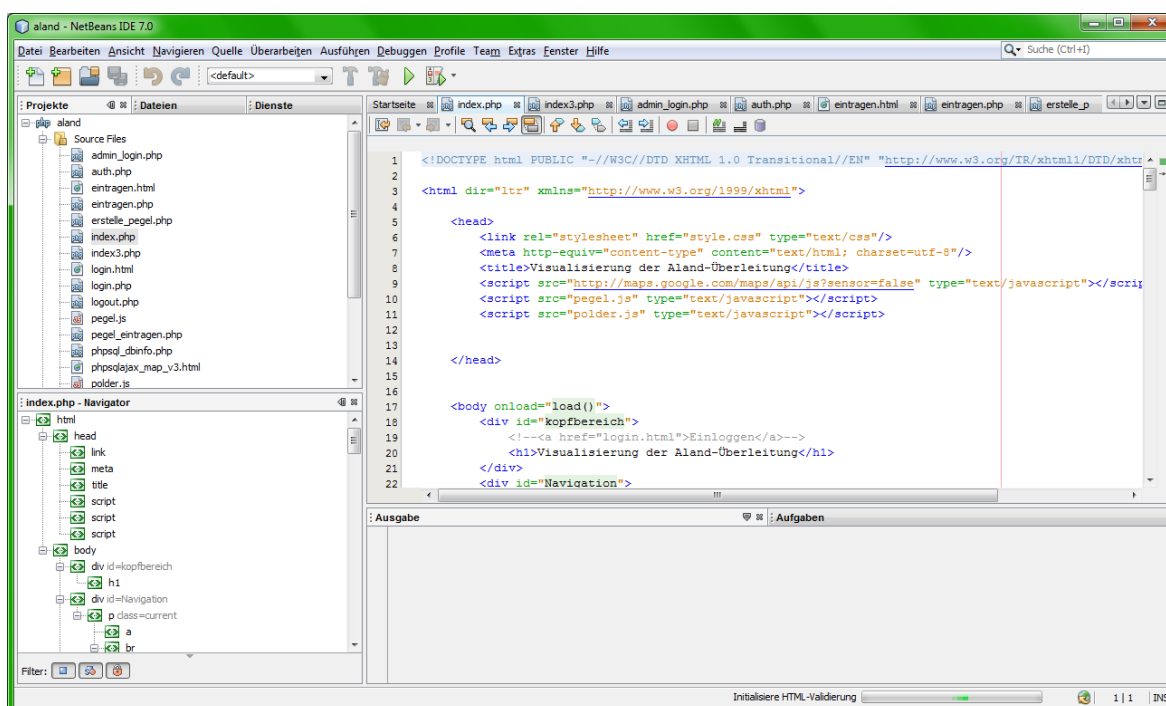


Abbildung 3-3 Ausschnitt aus der NetBeans IDE 7.0

3.1.3 XAMPP

XAMPP ist eine Zusammenstellung freier Software, die eine einfache Testumgebung des Apache Webservers, mit der Datenbank MySQL und den Skriptsprachen Perl und PHP bereitstellt. Damit kann das zu erstellende Web-Interface problemlos getestet werden. Zur Verwaltung der MySQL-Datenbank wird phpMyAdmin, eine freie PHP-Applikation verwendet, die eine Administration über den Web-Browser ermöglicht.

3.2 Technologien

3.2.1 PHP

PHP ist eine Skriptsprache, mit der hauptsächlich Webseiten dynamisch erzeugt werden. Es wurde vom Dänen Rasmus Lerdorf 1995 entwickelt und ursprünglich als „Personal Home Page Tools“ bezeichnet. Mit der Weiterentwicklung der Sprache hat sich auch der Name entwickelt, der nun „Hypertext Preprocessor“ lautet. Diese Definition beschreibt das eigentliche Verhalten des PHP-Interpreters.

„PHP unterscheidet sich von clientseitigen Sprachen wie JavaScript dadurch, dass der Code auf dem Server ausgeführt wird und dort HTML-Ausgaben generiert, die an den Client gesendet werden. Der Client erhält also nur das Ergebnis der Skriptausführung, ohne dass es möglich ist herauszufinden, wie der eigentliche Code aussieht.“ (The PHP Group, 2011)

Aber auch Ausgaben in andere Formate, wie z.B. PDF-Dokumente, sind möglich.

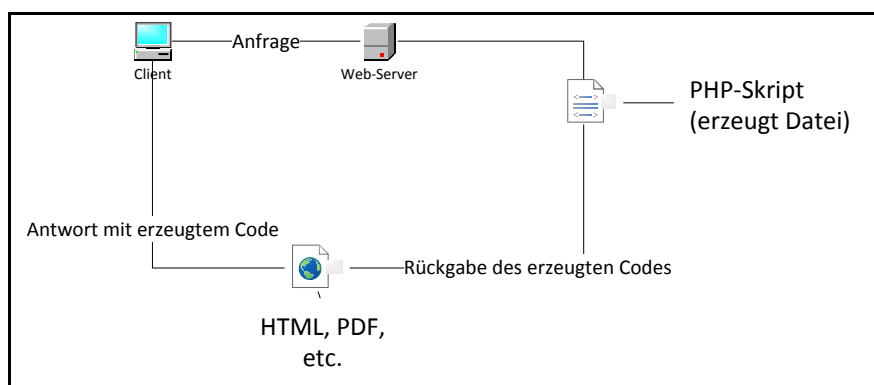


Abbildung 3-4 Darstellung der PHP-Funktionsweise

3.2.2 JavaScript

JavaScript ist eine Skriptsprache, die hauptsächlich beim Abrufen von HTML-Seiten zum Einsatz kommt. Dabei führt der Browser den JavaScript-Code aus und zeigt das Ergebnis an. Auch wenn es zu vermuten wäre, hat JavaScript mit der Programmiersprache Java nicht viel gemein.

3.2.3 MySQL

MySQL ist im Internet das am weitesten verbreitete Datenbanksystem im Open-Source-Bereich. KOFLER hat dafür viele Gründe:

- „MySQL ist schnell.
- MySQL ist stabil.
- MySQL ist leicht zu erlernen.
- MySQL läuft auf den gängigen Betriebssystemen (Windows, Linux, Mac OS X, diverse Unix-Derivate).
- MySQL-Anwendungen können in allen erdenklichen Programmiersprachen erstellt werden (z.B. in C, C++, C#, Java, Perl, PHP, Python, VB und VB.NET).
- MySQL ist im Internet umfassend dokumentiert, außerdem gibt es zahlreiche Bücher dazu.
- MySQL ist für viele Anwendungen kostenlos verfügbar (Lizenz GPL).
- Da die Lizenzbestimmungen der GPL ein Hindernis für manche kommerzielle Anwendungen sind, gibt es kostengünstige kommerzielle Lizenzen und optionale Support-Verträge.“ (Kofler, 2005, S. 23)

Vor allem im Bereich der Webservices kommt MySQL in Verbindung mit Apache und PHP zum Einsatz. Durch PHP werden SQL-Befehle ausgeführt, die zur Definition, Abfrage und Manipulation der Daten in der Datenbank verwendet werden. SQL, das als Abkürzung für „Structured Query Language“ steht, ist jene Datenbanksprache, die eine Reihe von Befehlen für die genannten Aktionen bereitstellt.

4 PLANUNG

Bei der Planung der Funktionalitäten des Hochwasserinformationssystems (HIS) muss geklärt werden, was das System leisten soll. Der Nutzer kann, anhand einer Karte oder tabellarisch, einen Überblick zur aktuellen Hochwasserlage erhalten, in dem Informationen zu einzelnen Bauwerken abgerufen werden, wie z.B. tagesaktuellen Pegelständen oder Wehrstatus. Außerdem kann der Wasserstand im Polder bei bestimmten Pegelständen visualisiert werden.

4.1 Geforderte Funktionalitäten des Hochwasserinformationssystems

Eine Funktion ist das Anzeigen der Bauwerksdaten in der Google Maps-Karte. Die MySQL-Datenbank enthält Informationen zu jedem Bauwerk. Im Einzelnen werden die Merkmale Name, zugehöriges Gewässer, Station, Nummer und die Koordinaten, in Längen- und Breitenangaben erfasst. Der aktuellste Wert aus der Datenbank, mit anderen Pegeldaten, wird auf der Google Maps-Karte durch Anklicken des geocodierten PINs dargestellt. Die Daten für einen bestimmten Pegel tabellarisch aufzulisten, gehört ebenfalls zum Funktionsbereich des Hochwasserinformationssystems. Eine weitere Funktion ist das Visualisieren der Überschwemmungsflächen in den Polderflächen Garbe und Wrechow. Das wird durch das Laden bestimmter KMZ-Dateien für den aktuellen Pegelwert realisiert. Diese KMZ-Dateien wurden im Vorhinein berechnet und werden statisch geladen.

Zum Einlesen/Eingeben neuer Pegelwerte, bzw. Aktionen am Wehr, sowie zum Erstellen neuer Hochwasserpegel innerhalb des Systems, wird ein Web-Interface benötigt, das diese Forderungen umsetzt. Dazu kann ein berechtigter Nutzer (Administrator) sich in den geschützten Bereich einloggen und diese Daten einlesen bzw. eingeben. Dabei werden Einträge in der Datenbank hinzugefügt.

Im Hochwasserinformationssystem sind zwei Benutzerrollen vorgesehen:

1. **Nutzer:** Der Nutzer hat die Möglichkeit sich das Hochwasserinformationssystem anzeigen zu lassen, und die aktuellen Daten abzurufen. Außerdem besteht die Möglichkeit, die Überschwemmungsflächen manuell, oder anhand des aktuellen Pegelwerts zu visualisieren.

2. Administrator: Der Administrator kann sich über ein Login-Formular einloggen und in den geschützten Bereich des Web-Interfaces eintreten. Danach kann er die aktuellen Pegelwerte oder Aktionen am Bauwerk anhand eines Formulars einlesen, oder diese händisch eingeben. Es besteht ebenso die Möglichkeit, dem System ein neues Bauwerk hinzuzufügen. Dafür wird ein gesondertes Formular aufgerufen, wo der Administrator die erforderlichen Werte eingibt, und die Position des Pegels mithilfe der vorliegenden Google Maps Karte angibt.

Anhand des nachfolgenden Modells (siehe Abb. 4-1) wird die Verhaltensweise des Hochwasserinformationssystems aus Sicht der Nutzer bzw. der Administratoren dargestellt:

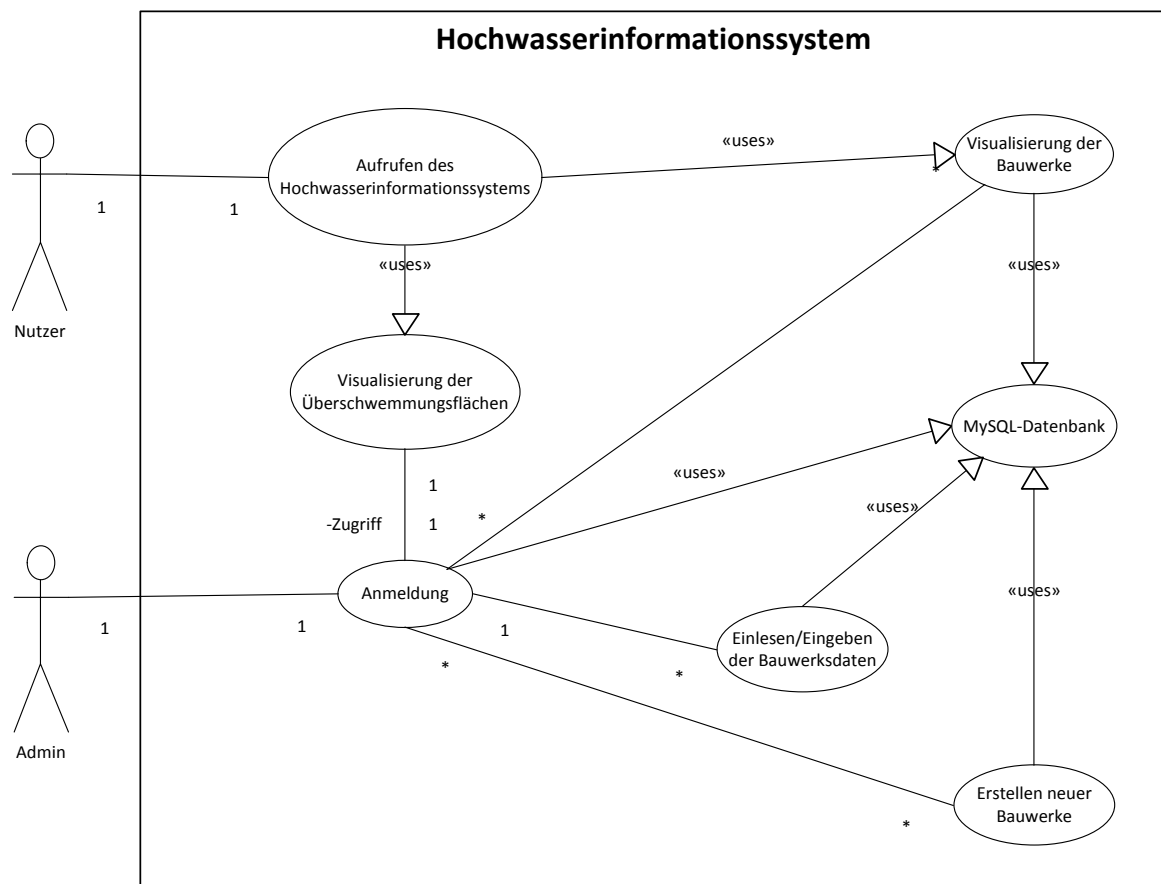


Abbildung 4-1 Benutzerrollenverteilung mit entsprechenden Aktionsmöglichkeiten

4.2 Datenbankentwurf

Bislang wurden die Anforderungen an das System abgesteckt und die Benutzerrollen definiert. Im folgenden Datenbankentwurf wird geklärt, wie die Daten im System, in der MySQL-Datenbank gespeichert werden.

Die Datenbank dient zum Speichern der wasserwirtschaftlichen Bauwerke mit deren dazugehörigen Daten. Dabei muss anfangs definiert werden, welche Daten für einen bestimmten Pegel benötigt bzw. gespeichert werden müssen. Aus diesen Erkenntnissen wird dann ersichtlich, welche Tabellen für die Datenbank benötigt werden, und welche Datentypen die einzelnen Attribute benötigen.

Zunächst werden intuitiv alle Angaben, die benötigt werden, im Rahmen eines einzigen Objekttyps *Bauwerk* dargestellt. Daraus entsteht in der Modellierung und dem daraus folgenden Normalisierungsprozess das *unnormalisierte Datenmodell*:

Bauwerk
<u>Name des Bauwerks</u>
<u>Gewässer des Bauwerks</u>
<u>Station des Bauwerks</u>
<u>Nummer des Bauwerks</u>
<u>Koordinaten des Bauwerks</u>
Wasserstand /Wehrstatus
Zeitpunkt des Wasserstands / Wehrstatus
Pegelnulldpunkt (PNP)

Abbildung 4-2 Unnormalisiertes Datenmodell für ein Bauwerk

Hieraus resultiert, dass jedes Bauwerk nur einfach vorkommen kann, es aber mehrere Werte für den Wasserstand, sowie die dem dazugehörigen Zeitpunkt geben kann. Auch die Koordinaten des Bauwerks benötigen mindestens zwei Eigenschaften, um eindeutig definiert zu sein. Sie weisen aktuell multiple Eigenschaften auf. Würde man dieses unnormalisierte Modell in ein Datenbank-

modell transformieren, würde es dazu führen, dass für jeden neuen Pegelwert ein komplett neuer Datensatz erzeugt würde, mit einigen Redundanzen. Außerdem belegt die mehrfache Sicherung identischer Daten nur unnötigen Speicherplatz. Der folgende Normalisierungsprozess dient dazu, diese Redundanzen zu entfernen.

1. Normalform

„Ein Datenmodell liegt in der 1. Normalform vor, wenn es keine multiplen Eigenschaften aufweist.“ (Jarosch, 2010, S. 94)

Das heißt die Koordinaten beispielsweise dürfen nicht als ein Attribut verwendet werden, sondern müssen in Breite und Länge (Latitude und Longitude) aufgeteilt werden:

Bauwerk
<u>Name</u>
<u>Typ</u>
<u>Gewässer</u>
<u>Station</u>
<u>Nummer</u>
<u>Geographische Breite (Latitude)</u>
<u>Geographische Länge (Longitude)</u>
Wasserstand des Pegels / Status des Wehrs
Zeitpunkt des Wasserstand / Wehrstatus
Pegelnullpunkt (PNP)

Abbildung 4-3 1. Normalform

2. Normalform

„Ein Datenmodell liegt in der 2. Normalform vor, wenn es sich in der 1. Normalform befindet und jede beschreibende Eigenschaft eines Objekttyps zwar vom Gesamtschlüssel, aber nicht bereits von einem Teilschlüssel dieses Objekttyps funktional abhängig ist.“ (Jarosch, 2010, S. 99)

Würden jetzt die Daten zum Bauwerk täglich aktualisiert werden, so würden einige Daten, wie Name des Bauwerks, Gewässer, Station, Nummer, sowie dessen geographische Koordinaten in jedem Datensatz stehen – die Daten wären also redundant. Dadurch besteht die Gefahr, dass die Integrität der Daten verletzt wird, d.h. Daten könnten unbeabsichtigt, absichtlich oder durch einen technischen Fehler verändert werden. Werden die Daten bei einem Datenbankeintrag verändert, so würde das nicht die anderen Einträge mit dem eigentlich gleichen Bezug betreffen. Die Lösung liegt in der Aufteilung der Tabelle; im Rahmen des Projekts in *bauwerk*, *pegel*, *wehr* und *gewaesser*.



Abbildung 4-4 Tabelle *bauwerk*

Die Tabelle *bauwerk* enthält somit nur noch Daten, die fest mit der *idbauwerk* zusammenhängen. Das Feld *idbauwerk* ist Primärschlüssel.

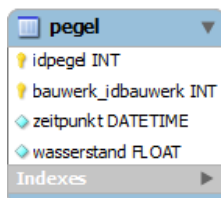


Abbildung 4-5 Tabelle *pegel*

Die Tabelle *pegel* enthält nur noch Felder, die von der *idpegel* und der *bauwerk_idbauwerk* abhängt. Es kann anhand des Fremdschlüssels *bau-*

werk_idbauwerk eindeutig ein Wert zugeordnet werden, mit entsprechendem Wasserstand und dazugehörigen Zeitpunkt des Wasserstands.

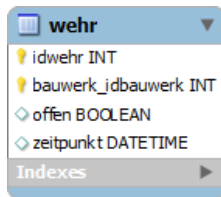


Abbildung 4-6 Tabelle wehr

Wird beim Erstellen des Bauwerks ein Wehr ausgewählt, so kann beim Ausführen von Aktionen am Wehr der Wehrstatus gesetzt werden, mit Angabe des Zeitpunkts.

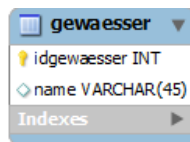


Abbildung 4-7 Tabelle gewaesser

Die Tabelle *gewaesser* enthält nur noch Felder, die von der *idgewaesser* abhängen.

Durch die Zerlegung in mehrere Tabellen sind die Redundanzen der Daten beseitigt.

Das Attribut *bauwerk_idbauwerk* aus der Tabelle *pegel* wird als Fremdschlüssel bezeichnet, der auf den Primärschlüssel der Tabelle *bauwerk* verweist. Das Attribut *gewaesser_idgewaesser* aus der Tabelle *bauwerk* wird ebenso als Fremdschlüssel bezeichnet, der auf den Primärschlüssel der Tabelle *gewaesser* verweist. Die Attribute *idpegel* und *bauwerk_idbauwerk* in der Tabelle *pegel* bilden den zusammengesetzten Primärschlüssel. Gleiches gilt für die Attribute *idwehr* und *bauwerk_idbauwerk* in der Tabelle *wehr*.

3. Normalform

„Ein Datenmodell liegt in der 3. Normalform vor, wenn es sich in der 2. Normalform befindet und keine beschreibende Eigenschaft eines Objekttyps von einer anderen beschreibenden Eigenschaft dieses Objekttyps funktional abhängig ist.“ (Jarosch, 2010, S. 104)

Da diese Anforderung bereits mit der zweiten Normalform erfüllt ist, kann das in der 2. Normalform gezeigte Modell, als 3. Normalform bezeichnet werden.

Im Folgenden wird nun die Datenbank des Hochwasserinformationssystems als Entity-Relationship-Modell dargestellt. Im ER-Modell werden Abhängigkeiten aus der realen Welt grafisch dargestellt. Es wird zwischen Objekten (*Entity*), ihren Beziehungen untereinander (*Relationship*) und den Informationen über die entsprechenden Objekte (*Attribute*) unterschieden.

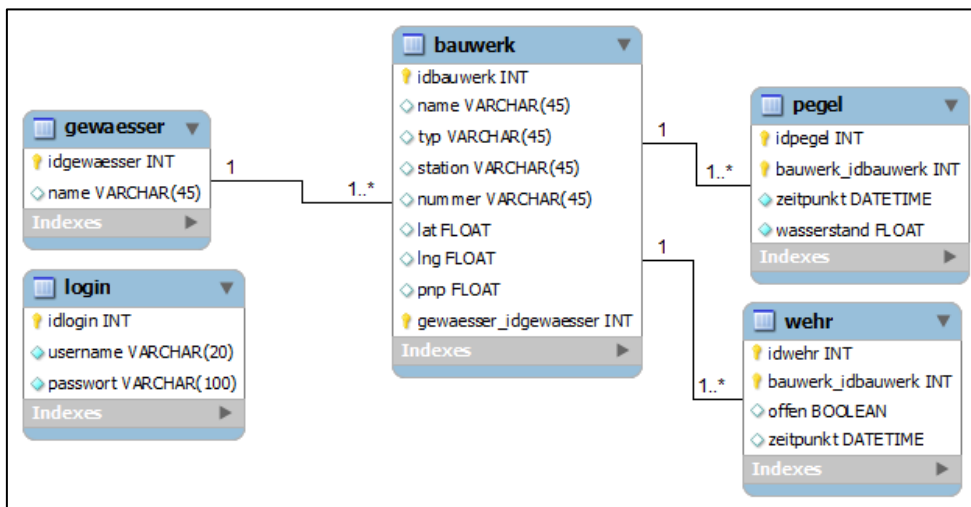


Abbildung 4-8 Entity-Relationship-Modell

Das ER-Modell besteht aus fünf Tabellen. Die Tabellen *pegel* und *wehr* sind aufgrund der Typisierung in *bauwerk* erstellt worden, und demzufolge bei mehreren Bauwerkstypen, mehr Typentabellen möglich. Vorerst soll sich das Hochwasserinformationssystem jedoch auf die Typen Pegel und Wehr beschränken. Vier dieser Tabellen stehen in Beziehung zueinander. Der Einsatz von Fremdschlüsseln ermöglicht die Verbindung der verschiedenen Tabellen. Die Tabelle *login* ist unabhängig und beinhaltet die Nutzer, die Zugriff zum geschützten Bereich des Systems haben.

Im Entity-Relationship-Modell befinden sich ausschließlich 1:n-Beziehungen. Zwischen der Tabelle *gewaesser* und der Tabelle *bauwerk* existiert die Verbindung, dass ein Bauwerk nur einem Gewässer zugeordnet werden kann, umgekehrt jedoch können zu einem Gewässer mehrere Bauwerke gehören. Gleiches gilt zwischen den Tabellen *bauwerk* und *pegel*: Jeder Pegelwert gehört

zu einem Bauwerk, jedoch kann und wird ein Bauwerk mehrere Pegelwerte besitzen. Analog die Beziehung zwischen *bauwerk* und *wehr*. Die Verbindung der Tabellen erfolgt über Fremdschlüssel, d.h. beim Abrufen von Daten in der Tabelle *bauwerk* verweist beispielsweise der Fremdschlüssel auf die ID des Gewässers in der Tabelle *gewaesser*, anhand dieser wird der Gewässername gelesen. Analog dazu wird beim Speichern neuer Pegelwerte in der Tabelle *pegel* die ID des dazugehörigen Bauwerks mitgespeichert. Sollen nun Werte eines bestimmten Pegels angezeigt werden, so wird in der Tabelle *pegel* nach allen Werten mit der dazugehörigen Bauwerks-ID, die dort als Fremdschlüssel gespeichert werden, gesucht und angezeigt. Sollte nun einmal z.B. der Name eines Gewässers geändert werden, so geschieht dies ausschließlich in der Tabelle *gewaesser*. Durch das Referenzieren auf diese Tabelle sind nun alle Datensätze aktuell und tragen den neuen Namen des Gewässers.

In dem ER-Modell sind auch die Datentypen der einzelnen Attribute festgelegt. Die Primärschlüssel und Fremdschlüssel der einzelnen Tabellen sind als Ganzzahlenwert (Integer) definiert. Die Koordinaten der Bauwerke (Latitude und Longitude) sowie der Pegelnullpunkt (PNP) sind Gleitkommazahlen, woraus der FLOAT-Datentyp resultiert. Das Attribut *zeitpunkt* in den Tabellen *wehr* und *pegel* wird als Datum inklusive Uhrzeit gespeichert, wofür MySQL den Datentyp DATETIME vorsieht. MySQL ruft DATETIME-Werte im Format 'YYYY-MM-DD HH:MM:SS' (Jahr-Monat-Tag Stunden-Minuten-Sekunden) ab und stellt das Format auch in dieser Form dar. Der Wert *offen* in der Tabelle *wehr* wird als Boolean definiert, da nur die Angabe erforderlich ist, ob das Wehr-Bauwerk geöffnet oder geschlossen ist. Alle sonstigen Werte werden als Zeichenwerte im Typ VARCHAR gespeichert.

5 TRANSFORMATION VON RASTERDATEN NACH KML

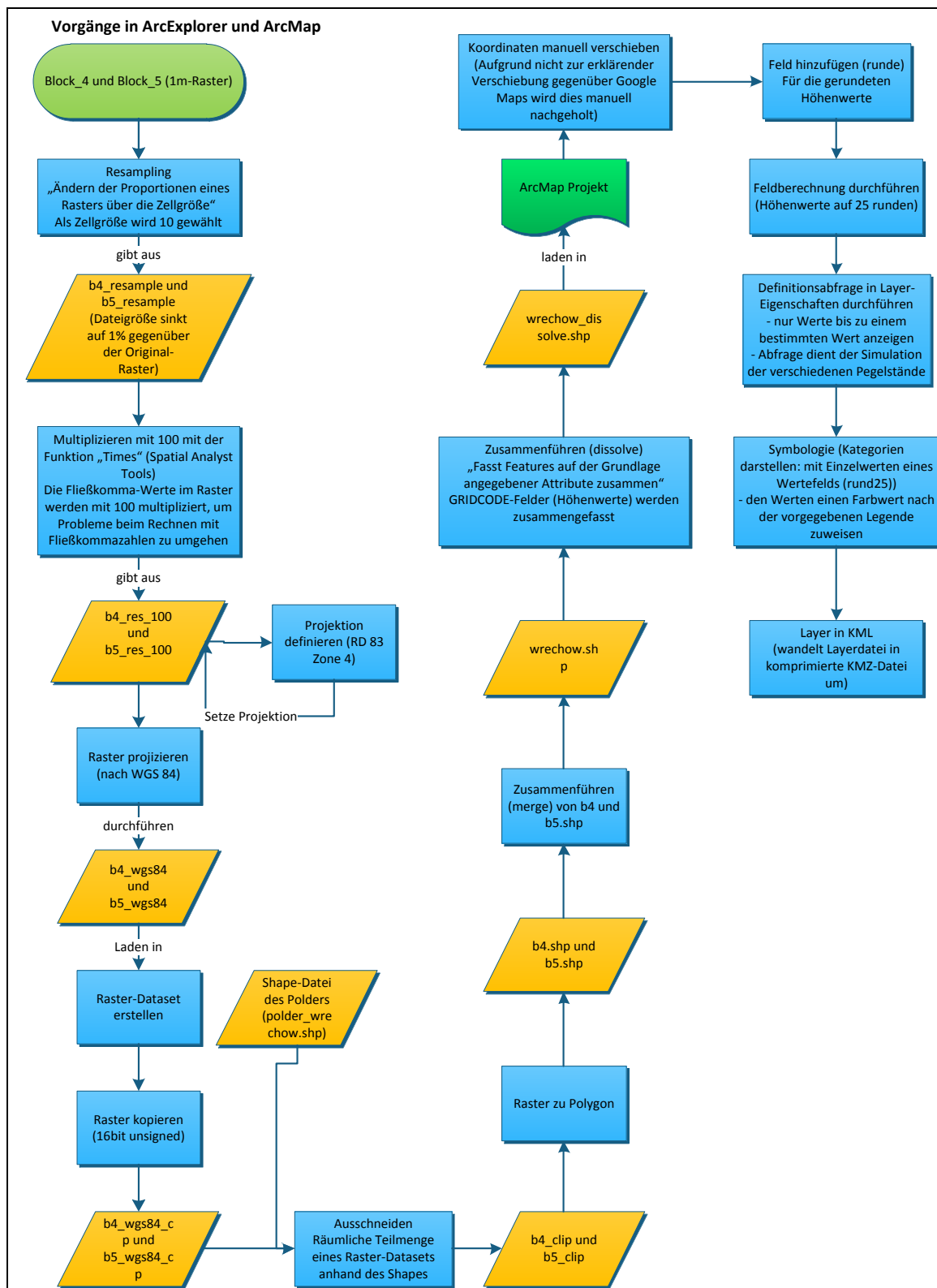
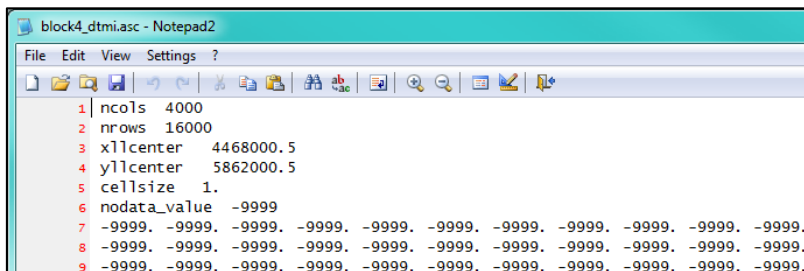


Abbildung 5-1 Schema Raster zu KML am Beispiel Polder Wrechow

Als Datengrundlage stand im Rahmen des Projekts ein Digitales Geländemodell (DGM) im ASCII GRID-Format zur Verfügung. Da eine direkte Visualisierung des DGM mithilfe der Google Maps API nicht möglich ist, musste dieses anhand vieler auszuführender Schritte, in den ESRI-Programmen ArcCatalog und ArcMap, in das Google-Format Keyhole Markup Language (KML) gewandelt werden. Diese Schritte werden im obigen Modell (siehe Abb. 5-1) dargestellt und in den folgenden Kapiteln erläutert.

5.1 Verarbeitung der Rasterdaten

Die vorliegenden Rasterdaten wurden aus einer ASCII-Punktdatei errechnet.



```
block4_dtmi.asc - Notepad2
File Edit View Settings ?
1 | ncols 4000
2 | nrows 16000
3 | xllcenter 4468000.5
4 | yllcenter 5862000.5
5 | cellsize 1.
6 | nodata_value -9999
7 | -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999.
8 | -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999.
9 | -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999. -9999.
```

Abbildung 5-2 ASCII Grid-Datei

ASCII GRID besteht aus einem Header und den Höhenwerten des GRIDs (siehe Abb. 5-2). *ncols* und *nrows* geben die Anzahl der Spalten und Zeilen an. *xllcenter* und *yllcenter* geben das Zentrum des Rasters an. *cellsize* bezieht sich auf die Auflösung des Rasters. Die Werte *nodata_value* werden als fehlend oder nicht zutreffend bezeichnet.

Da das vorliegende ASCII GRID eine Auflösung besitzt die höher ist, als im Rahmen des Projekts benötigt, wurde eine Komprimierung des Rasters vorgenommen. Um die vorliegende Rasterdatei zu komprimieren, wurde die Funktion *Resampling* eingesetzt. Die Proportionen des Raster-Datasets werden über die Zellgröße verändert. Der Wert wurde mit 10 belegt, woraus eine Größe des neuen Rasters von 1% gegenüber dem Original-Raster resultiert.

Um das Raster in ein Polygon umzuwandeln, werden ganzzahlige Werte benötigt. Die Höhenwerte des Rasters liegen jedoch im Fließkommazahlenbereich vor und werden deshalb mit dem Faktor 100 multipliziert. Dazu dient die Spatial Analyst Funktion *Multiplizieren*.

Die vorliegenden Rasterdateien stammen aus einer Laserscan-Befliegung und wurden ursprünglich in Blöcke unterteilt, die jeweils einen Streifen des Projektgebiets abdecken. Da die Ausdehnung des Polders die Breite eines Streifens überdeckt, sind zwei Blöcke notwendig um die gesamte Polderfläche abzufassen (siehe Abb. 5-3)

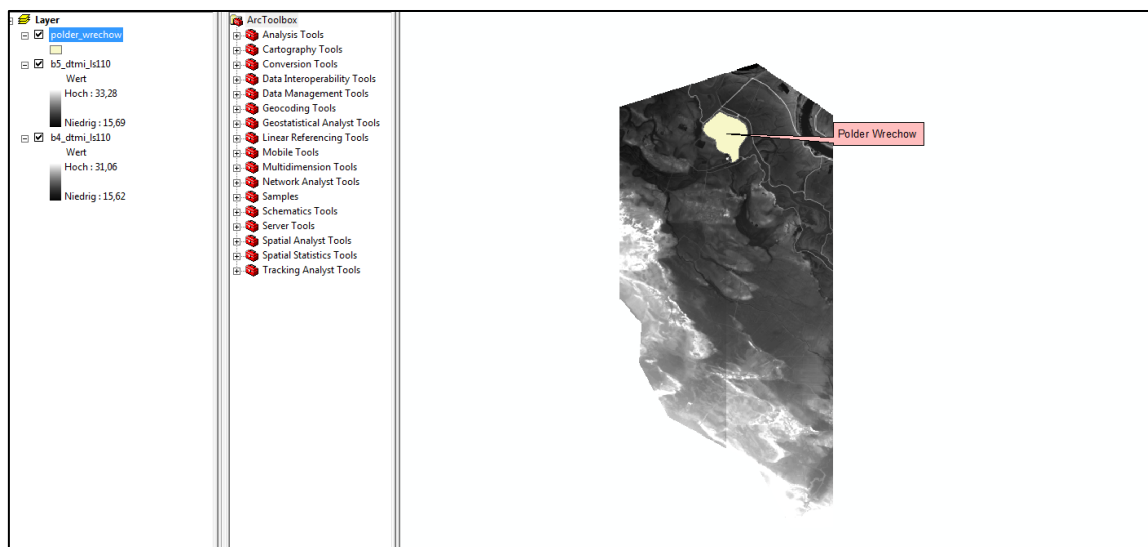


Abbildung 5-3 Ansicht der Rasterstreifen (Blöcke) mit Shape des Wrechow-Polders (Ausschnitt ArcMap)

Um aus dem kompletten Streifen des Rasters nur den gewünschten Ausschnitt des Polders zu erhalten, kommt die Funktion *Ausschneiden* zum Einsatz. Es wird der gesamte Rasterstreifen mit der Shapedatei des Polders verschnitten.

5.2 Transformationen zwischen den Koordinatensystemen

Das ursprüngliche Raster liegt im geodätischen Bezugssystem RD/83, „das Ergebnis einer Transformation vom 42/83 in das Rauenbergdatum“ (GeoSN) ist, vor. Referenzellipsoid ist das Bessel-Ellipsoid von 1841. Um die Daten auf Google Maps zu projizieren, muss eine Koordinatentransformation ins Weltkoordinatensystem WGS 84 durchgeführt werden.

In ArcExplorer wird mithilfe der Funktion *Projektion definieren* das Koordinatensystem „RD83 Zone 4 C“ gewählt. Diese Projektion ergibt bei Transformation nach WGS 84 die bestmögliche Übereinstimmung. Die Funktion *Raster projizieren* erlaubt, die Rasterdaten in das World Geodetic System 84 zu projizieren.

5.3 Ausgabe der Daten in KML - Dateien

Die projizierten Rasterstreifen werden mit der Funktion *Raster zu Polygon* in Polygon-Shapedateien umgewandelt, die durch die *Merge*-Funktion zu einem Polygon-Shape zusammengefügt werden. Die Attribut-Tabelle des Features beinhaltet alle einzelnen Höhenwerte, d.h. jedem Punkt ist ein Höhenwert zugewiesen. Da dies nicht benötigt wird, können die Features auf der Grundlage der Höhenwerte mit der *Dissolve*-Funktion zusammengefasst werden, d.h. zu jedem Höhenwert gehört eine Fläche, die aus mehreren Teilen besteht. Das zusammengeführte Shape wird dann in ein ArcMap-Projekt geladen. Die Shapedatei kann mit der Funktion *Layer in KML* in eine KML-Datei exportiert werden, wobei ArcMap voreingestellt eine KMZ, eine komprimierte KML-Datei, erstellt. Wird diese KMZ-Datei mit Google Earth betrachtet, so kann eine unerklärliche Verschiebung beobachtet werden. Zur Auflösung der Verschiebung wurde das Feature (Polder Wrechow) im Editor-Modus in ArcMap mittels manueller Verschiebung um einen Wert (x, y) bewegt, was als hinreichend genau betrachtet wird.

Für die Visualisierung wird eine Abstufung der Höhenwerte in 25 cm-Abständen als ausreichend angesehen. Dafür wird der Attributtabelle des Features eine Spalte (*runde*) hinzugefügt, in der die gerundeten Werte auf 25 cm eingetragen werden. Dazu wird die Feldberechnung in ArcMap mit einer Formel durchgeführt, die alle Werte auf 25 rundet

$$\text{runde} = \text{Fix} ([\text{GRIDCODE}]/25) * 25 + 25$$

Die Funktion *FIX* gibt den ganzzahligen Wert zurück, *GRIDCODE* ist die nicht gerundete Geländehöhe

Folgende Beispieltabelle verdeutlicht dies:

Wertebereich	Gerundet (25)
z.B. 1700 - 1724	1725
z.B. 1725 – 1749	1750
z.B. 1750 – 1774	1775
z.B. 1775 – 1799	1800

Tabelle 1 Wertebereich und Rundungswerte auf 25

Um die verschiedenen Wasserstände in den Poldern zu simulieren, wird in den Layer-Eigenschaften des Shapes eine Definitionsabfrage durchgeführt, z.B.

```
SELECT * FROM wrechow_dissolve WHERE: "runde" <= 1725
```

Es werden im Beispiel alle Werte angezeigt die kleiner gleich 1725 cm sind. Die Symbologie des Layers wird in Kategorien dargestellt, mit Einzelwerten eines Wertefelds (runde). Nach Klick auf „Alle Werte hinzufügen“ erscheinen drei Werte (1675, 1700, 1725) mit unterschiedlicher Farbgebung, die nach vorgegebener Legende anhand der Wassertiefe farblich dargestellt werden.

Dieser Ausschnitt wird über die Funktion *Layer in KML* in eine KMZ-Datei umgewandelt und gespeichert. Dieser Vorgang wird für alle weiteren Werte wiederholt, bis alle Wasserstände simuliert wurden.

Die KMZ-Dateien werden dann auf einem Webserver platziert, wo sie bei Funktionsaufruf des Web-Interfaces abgerufen und dargestellt werden.

6 ENTWICKLUNG EINES PROTOTYPEN

In dem folgenden Kapitel soll die Entwicklung des zuvor beschriebenen Entwurfs gezeigt werden. Das Ergebnis ist der Prototyp des Hochwasserinformationssystems.

Für das Web-Interface sind fünf hauptsächliche Aufgabenbereiche zu realisieren: Basierend auf der Einbindung der Google Maps API, ist das Zeigen der Bauwerke mit den jeweiligen Detailinformationen, wie Wasserstände beim Pegel oder dem Zustand des Wehrs innerhalb der Google Maps API erforderlich. Weiterhin soll der Nutzer die Flächen bei definierten Wasserständen in den Poldern Garbe und Wrechow visualisieren können. Daneben gilt es, dem Administrator einen geschützten Bereich zur Verfügung zu stellen, in welchem die Möglichkeit gegeben ist neue Bauwerke anzulegen, sowie aktuelle Daten zu den Pegeln bzw. Wehren entweder manuell, oder mittels eines Imports einzugeben bzw. einzulesen. Weiterhin hat er die Möglichkeit zur Pflege der Datenbank, alte Datensätze die älter als einen Monat oder einen Tag sind, zu löschen.

Im Hintergrund arbeitet eine MySQL-Datenbank, die im vorherigen Kapitel entworfen wurde und nun zuvor implementiert wird.

6.1 Die MySQL-Datenbank

Das Datenbank-Modell wurde mithilfe des *MySQL Workbench 5.2 CE*, einem Programm zur visuellen Datenbankmodellierung, erstellt und über phpMyAdmin an die Datenbank übergeben. Der Vorteil in diesem Programm liegt darin, dass das Modell graphisch erstellt werden kann. Die Tabellen werden erstellt und die Attribute mit den entsprechenden Datentypen hinzugefügt, sowie die Relationen zwischen den Tabellen definiert. Daraufhin können die Tabellen in der Datenbank über eine Export-Funktion erstellt werden.

6.2 Architektur und Struktur des Web-Interfaces

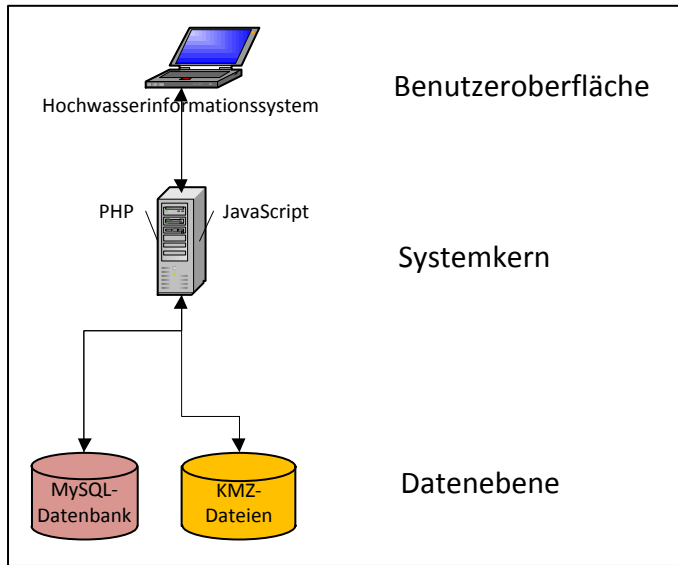


Abbildung 6-1 Architektur des Web-Interfaces

Für die Architektur des Web-Interfaces wird ein Client-Server-Modell verwendet. Es besteht aus der Datenebene, die die MySQL-Datenbank als auch KMZ-Dateien beinhaltet; aus dem Systemkern, der die Eingaben des Anwenders verarbeitet und die Datenbank als auch die KMZ-Dateien an den Systemkern anbindet; sowie aus der Benutzeroberfläche, die das Web-Interface darstellt.

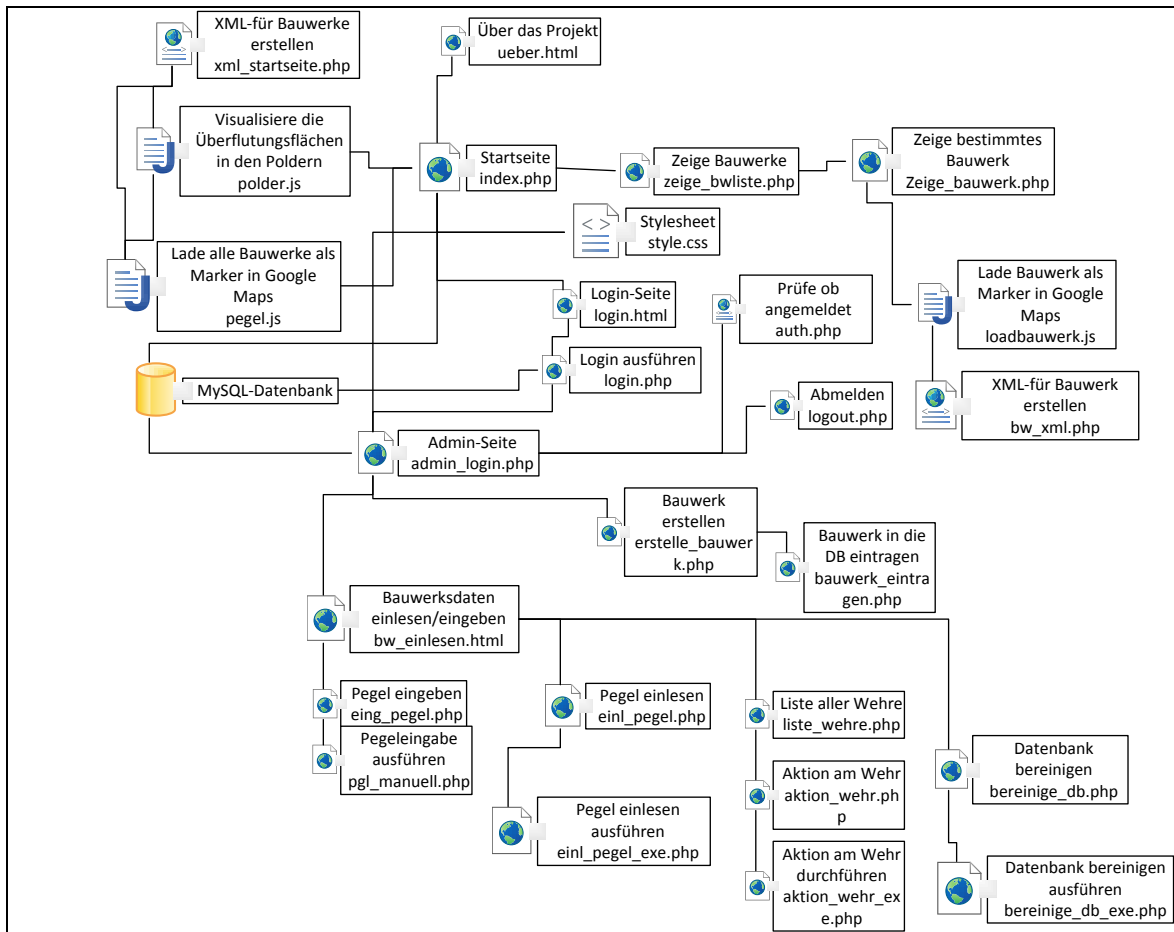


Abbildung 6-2 Struktur des Web-Interfaces

Die Abbildung 6-2 zeigt den schematischen Aufbau des Web-Interfaces als Modell, d.h. es wird der Zugriff der Seiten auf andere dargestellt. Der Vorteil dieses Modells ist eindeutig: Es dient dem Entwickler zur Übersicht beim Programmieren und kann außerdem einfach erweitert werden.

Alle durch Linien verbundenen Seiten oder Skripte greifen entweder durch Benutzerinteraktionen oder durch interne Funktionen nach Benutzereingaben aufeinander zu. Die Linien geben den Weg an, der nötig ist, um auf eine bestimmte Seite zu gelangen.

6.3 Interaktion zwischen PHP-Seiten

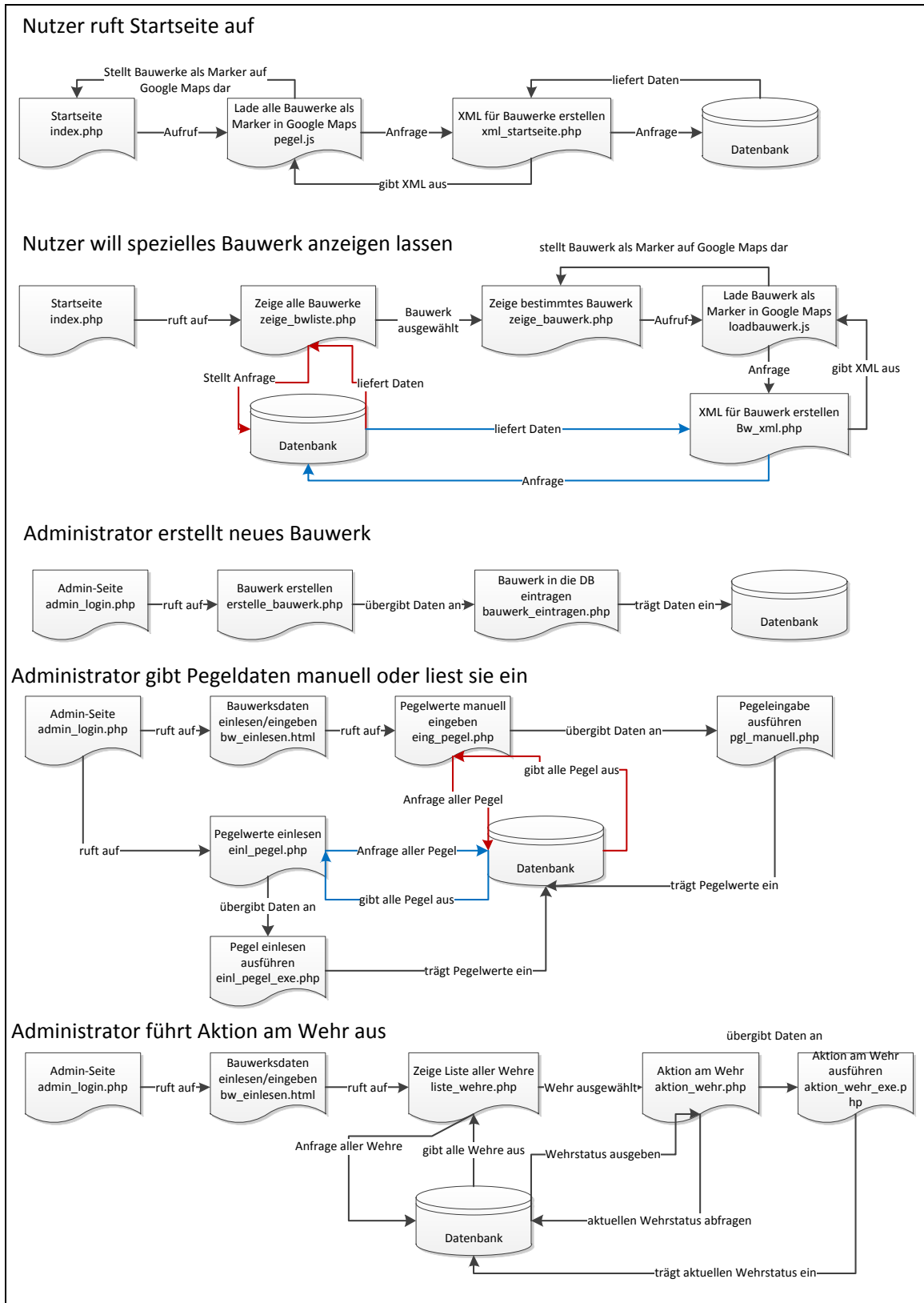


Abbildung 6-3 Schematischer Ablauf einiger Aktionen im Web-Interface

Die Abbildung 6-3 zeigt einen Ausschnitt des Ablaufs einiger Aktionen, die im Prototyp möglich sind. Es wird gezeigt, wie die PHP-Seiten miteinander interagieren und welche Aktionen durch diese ausgeführt werden. Nähere interne Funktionen spielen in dem Schema jedoch keine Rolle.

6.4 Entwurf eines Layouts

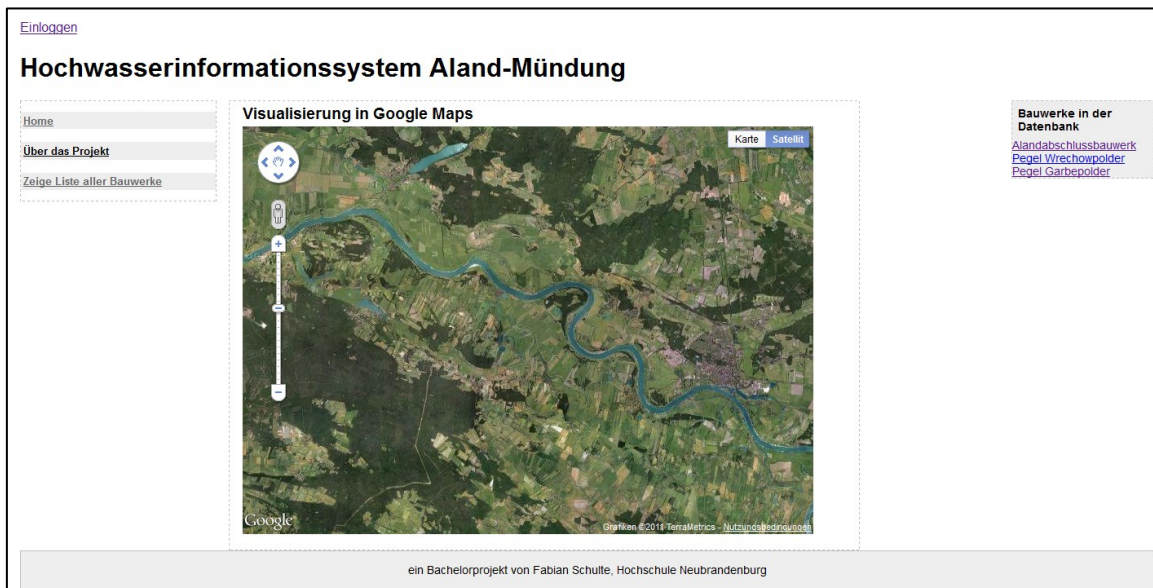


Abbildung 6-4 Layout-Entwurf der Seite *Zeige Liste aller Bauwerke*

Das Web-Interface wurde neben einem Kopf- und einem Fußbereich dreispaltig gestaltet. Dazu gehört der linke Navigationsbereich, der Hauptteil im Zentrum und eine rechte Spalte (siehe Abb. 6-4).

Der Kopfbereich enthält die Überschrift des Projekts, bzw. der aktuell geöffneten Seite, sowie einen Verweis auf ein Login-Formular, das nach erfolgreichem Login Zugriff auf den Administrationsbereich der Seite gewährt. Der Navigationsbereich erlaubt es dem Nutzer zu den verschiedenen Seiten bzw. Funktionen zu wechseln. Der Hauptteil beinhaltet das Google Maps API, und dient dem Eingeben bzw. Anzeigen von verschiedenen Daten. Zur rechten Spalte gehört eine Übersicht der Projekte in Listenform; auf der Startseite eine Möglichkeit zum Eingeben manueller Pegelstände in Bezug auf die Polderflächen und eine Legende der Wassertiefen.

Das Aussehen spezieller Elemente, wie die Schriftarten, Farben und Formen der verschiedenen Schrifttypen, die Farben der Links, die Größe der Container, die die

Bereiche umschließen wurden in einer separaten Datei, einer CSS-Datei, festgelegt. CSS (*Cascading Stylesheets*), ist eine Ergänzung zu HTML, mit der HTML-Elemente eindeutig positioniert und formatiert werden können. Die CSS-Datei kann bequem in jede einzelne Seite eingebunden werden und erzeugt somit ein einheitliches Layout.

6.5 Administrator-Bereich

Wie bereits in Kapitel 4.1 erwähnt, gibt es einen geschützten Bereich in dem Hochwasserinformationssystem, der es dem Administrator ermöglicht das HIS zu verwalten, bzw. zu erweitern. Dazu muss sich die entsprechende Person über den Menüpunkt „Einloggen“ im System mit Benutzernamen und Passwort anmelden (siehe Abb. 6-5).



Hochwasserinformationssystem Aland

Home
Über das Projekt
Admin-Bereich
Bauwerk erstellen
Bauwerksdaten eingeben/einlesen
Zeige Bauwerksliste

Geschützter Bereich

Dein Username:

Dein Passwort:

Login

ein Bachelorprojekt von Fabian Schulte, Hochschule Neubrandenburg

Abbildung 6-5 Login-Formular

Nach Eingeben der Benutzerdaten wird in der MySQL-Datenbank eine Abfrage an die *login*-Tabelle gestellt. Das Passwort des Nutzers wurde beim Erstellen mithilfe von *md5*, eine Funktion die jede beliebig Nachricht in einen 128-bit Hashwert (Prüfsumme) umwandelt, der in Form eines 32-stelligen Hexadezimalwerts in der Passwort-Spalte gespeichert wird. Aus der Nachricht „Hallo Welt“ wird z.B. „5c372a32c9ae748a4c040ebadc51a829“ nach der Umwandlung.

Der Benutzer wird in der Tabelle *login* gesucht, und das übergebene Passwort mit dem sich in der Datenbank befindenden, dem Nutzer zugeordneten Passwort, verglichen. Ist der Vorgang erfolgreich, wird der Session (Sitzung) ein Benutzer zugewiesen und diese auf *angemeldet* gesetzt.

Beim Aufrufen von geschützten Seiten wird zuvor eine weitere PHP-Seite (*auth.php*) aufgerufen, die überprüft ob die Sessionvariable *angemeldet* noch

existiert und diese auf *true* gesetzt ist – kurz, ob die Sitzung noch aktiv ist. Ist dies nicht der Fall wird auf das Login-Formular verwiesen.

Ist der Anmeldevorgang erfolgreich wird die Seite *admin_login.php* aufgerufen (siehe Abb. 6-6).



Abbildung 6-6 Geschützter-Bereich

Der geschützte Bereich bietet verschiedene Aktionen an, die dem Administrator vorbehalten sind. Diese sollen im Folgenden beschrieben werden.

6.5.1 Anlegen eines Bauwerks

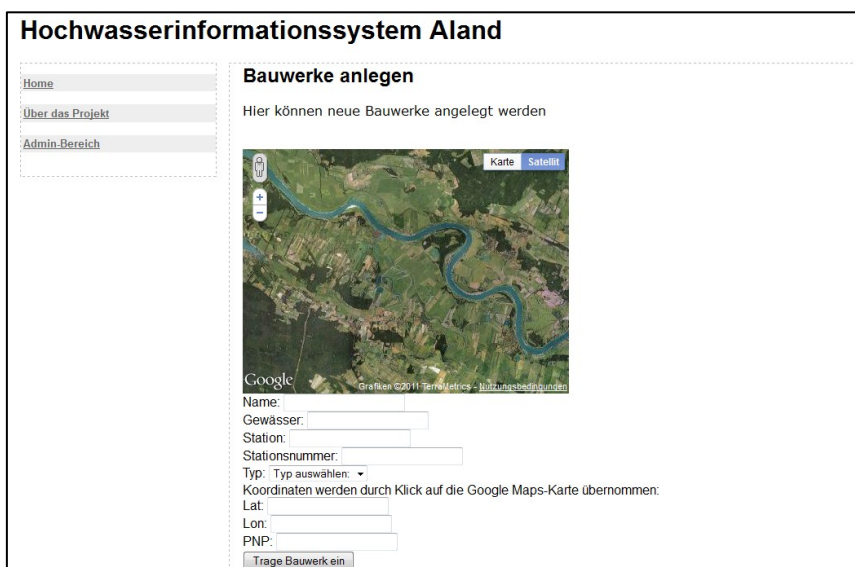


Abbildung 6-7 Anlegen eines Bauwerks

Die Seite „Bauwerke anlegen“ (siehe Abb. 6-7) beinhaltet ein Formular, das dem Administrator ermöglicht, dem Hochwasserinformationssystem ein neues Bauwerk hinzuzufügen. Hierzu werden die Daten zum Bauwerk in die entsprechenden Felder eingegeben, der Marker auf der Google Maps Karte wird per Maus positioniert. Die Koordinaten (Latitude und Longitude) werden dabei automatisch mithilfe einer JavaScript-Funktion an die Textfelder „Lat“ und „Lon“ übergeben. Über den Button wird ein neues Bauwerk in Datenbank geschrieben.

6.5.2 Einlesen / Eingeben von Daten zu einem Bauwerk

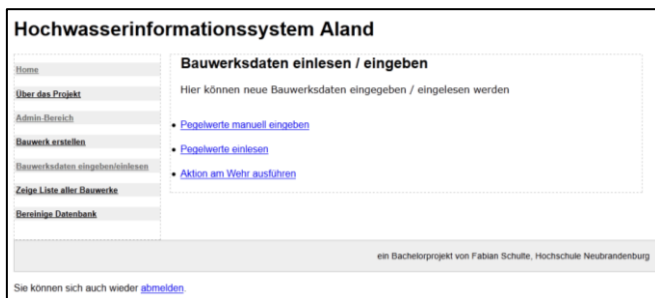


Abbildung 6-8 Bauwerksdaten einlesen / eingeben

Die Seite „Bauwerksdaten einlesen / eingeben“ (siehe Abb. 6-8) bietet Wege an, Daten der Bauwerke zu aktualisieren oder Aktionen daran auszuführen. Zum einen können neue Pegelwerte manuell oder über ein Formular nach Import aus dem Excel-Format *x/s* eingegeben bzw. eingelesen werden. Als Aktion am Bauwerk wird das Schließen oder Öffnen von Wehren bezeichnet. Im Folgenden soll auf die drei beschriebenen Möglichkeiten eingegangen werden.

6.5.2.1 Pegelwerte manuell eingeben



Abbildung 6-9 Pegelwert manuell eingeben

Sollen die Pegelwerte manuell eingegeben werden, so wird ein Formular aufgerufen, das eine Dropdown-Liste bereitstellt, in der alle Pegel aufgelistet sind. Daraus wird der zu aktualisierende Pegel ausgewählt, sowie in den entsprechenden Textfeldern der Pegelwert und das Datum inklusive Uhrzeit eingegeben (siehe Abb. 6-9). Durch Absenden des Formulars werden die Daten in das HIS eingetragen.

6.5.2.2 Pegelwerte aus Excel einlesen

Abbildung 6-10 Formular Pegelwerte aus Excel einlesen

	A	B
1	Parameter	W
2	Station	KJ Wanzer OP
3	Gewässer	Aland
4	Stationsnummer	594050
5		
6	21.08.2011 05:45	200
7	21.08.2011 06:00	200
8	21.08.2011 06:15	200
9	21.08.2011 06:30	200
10	21.08.2011 06:45	200
11	21.08.2011 07:00	200
12	21.08.2011 07:15	200
13	21.08.2011 07:30	200
14	21.08.2011 07:45	200
15	21.08.2011 08:00	200

Abbildung 6-11 Exportierte Excel-Tabelle

Eine weitere Möglichkeit Pegeldata einzulesen, ist der Import aus einer Excel-Tabelle. Das Programm WISKI hat die Möglichkeit die aufgenommenen Pegeldata nach Excel zu exportieren. Die Spalten mit dem Datum, inklusive Uhrzeit sowie der Wasserstand werden in Excel markiert (siehe Abb. 6-11) und kopiert, und in ein mehrzeiliges Eingabefeld (*textarea*), im Formular „Pegelwerte einlesen“ (siehe Abb. 6-10) eingefügt.

Nach Absenden des Formulars geschieht folgendes: Die eingegebenen Daten werden so ausgelesen und verarbeitet, dass sie als einzelne Datensätze in die Datenbank geschrieben werden können. Dazu wird zuerst der gesamte String in der eingelesenen *textarea* durch die PHP-Funktion *explode* getrennt, sofern ein Zeilenumbruch vorliegt. Die Ergebnisse werden in einem *Array* gespeichert. Die einzelnen Elemente des Arrays werden dann durch weitere *explode*-Funktionen getrennt und später wieder zusammengefügt, dass das Datum in der Form *YYYY-MM-DD HH:MI:SS* (Jahr-Monat-Tag Stunden:Minuten:Sekunden) im DATETIME-Format und der Pegelwert, in der MySQL-Datenbank gespeichert werden kann. Um die Datenbank nicht unnötig zu überfüllen, wurde eine Funktion eingebaut, die nur Werte mit voller Stunde (00 Minuten), an die Datenbank übergibt.

6.5.2.3 Aktion an einem Wehr durchführen

Soll an einem Bauwerk vom Typ Wehr eine Aktion durchgeführt werden, so ist dies auch durch das Hochwasserinformationssystem möglich. Beim Aufrufen der Seite „Aktion am Wehr durchführen“ wird zunächst eine Seite aufgerufen, die alle verfügbaren Wehre auflistet. Wird dann ein bestimmtes Wehr ausgewählt, kann die Aktion daran „ausgeführt“ werden, in dem der Wehrstatus verändert wird (siehe Abb. 6-12).

Abbildung 6-12 Aktion am Wehr durchführen

6.5.3 Datenbank bereinigen

Es besteht die Möglichkeit die Datenbank der Pegelwerte zu bereinigen. Dafür sind dem Administrator Möglichkeiten gegeben, Einträge in der Datenbank, die älter als einen Monat oder älter als einen Tag sind, zu löschen (siehe Abb. 6-13). Dafür wird eine MySQL-Abfrage ausgeführt, die alle Einträge auflistet, die älter als einen Monat bzw. einen Tag sind, und diese aus der Datenbank, bei Klick auf den Link, entfernt.

Abbildung 6-13 Datenbank bereinigen

6.6 Die Visualisierung der Bauwerke

Um die Bauwerke nicht nur in Textform anzuzeigen, wird eine Darstellung in Kartenform angestrebt. Dafür werden die Bauwerke der Google Maps API als Marker auf der Karte dargestellt. Wie bereits beschrieben wird im Rahmen der Bachelorarbeit die „Google Maps JavaScript API V3“ verwendet. Diese zeichnet sich gegenüber seinem Vorgänger, der Version 2, nicht nur dahingehend aus, dass für die Erstellung einer eigenen Karte kein sogenannter API-Key bei Google mehr beantragen werden muss, sie wurde auch „für eine kürzere Ladezeit konzipiert, insbesondere bei mobilen Browsern, zum Beispiel auf Android-basierten Geräten und auf dem iPhone™.“ (Google Inc., Google Maps JavaScript API V3 – Allgemeines, 2011)

Google hat eigens ein Tutorial geschrieben, das Entwicklern zeigt, wie Google Maps in Verbindung mit einer MySQL-Datenbank verwendet wird. Es wurde für die Entwicklung des Prototyps angepasst und verwendet. Dies soll in den folgenden Schritten beschrieben werden:

1. Schritt Die Tabelle erstellen

Das Aussehen und die Form der MySQL-Tabelle wurden bereits in einem früheren Kapitel beschrieben. Die erforderliche MySQL-Tabelle ist vorhanden.

2. Schritt Die Tabelle füllen

Auch das Füllen der Tabelle wurde bereits beschrieben. Es erfolgt sowohl in dem Schritt *Anlegen eines Bauwerks* (siehe Kap. 6.5.1), als auch im Bereich *Einlesen / Eingeben von Daten zu einem Bauwerk* (siehe Kap. 6.5.2).

3. Schritt XML mit PHP ausgeben

Um die Daten der MySQL-Tabelle auszugeben, wird empfohlen diese zuvor in eine XML-Datei auszugeben und diese als „Brücke zwischen Datenbank und Google- Karte“ (Google Inc., PHP/MySQL mit Google Maps verwenden, 2011) zu nutzen, um die Ladezeiten zu verkürzen. Im Projekt wird u.a. beim Aufrufen der Startseite, die JavaScript-Funktion `load()` aufgerufen, die die Seite `xml_startseite.php` intern aufruft, und den darin erzeugten XML-Code ausgibt. Die Seite `xml_startseite.php` erstellt den Hauptknoten vom Typ „markers“ und baut eine Datenbankverbindung zur Tabelle `bauwerk` auf. In dieser Tabelle werden alle Zeilen durchlaufen und zu jeder Zeile (Bauwerk) wird ein XML-Knoten, mit den erforderlichen Attributen, erstellt und an den Hauptknoten angehängen. Der manuelle Aufruf der `xml_startseite.php` im Projekt zeigt zum Beispiel folgenden Ausschnitt:

```
Mit dieser XML-Datei sind anscheinend keine Style-Informationen verknüpft. Nachfolgend wird die Baum-Ansicht des Dokuments angezeigt.

<markers>
  <marker id="15" name="Alandabschlussbauwerk" typ="Wehr" station="Alandabschlussbauwerk" nummer="548996" lat="53.0183" lng="11.5878" pup="0" offen="0"
zeitpunkt_vhr="2011-08-24 12:00:00"/>
  <marker id="16" name="Pegel Wrechowpolder" typ="Pegel" station="Wrechowpolder" nummer="789645" lat="53.0175" lng="11.5881" pup="160" zeitpunkt_pgl="2011-08-21
08:00:00" wasserstand="200" offen="0" zeitpunkt_vhr="2011-08-24 12:00:00"/>
  <marker id="17" name="Pegel Garbepolder" typ="Pegel" station="Garbepolder" nummer="478965" lat="53.0177" lng="11.5904" pup="160" zeitpunkt_pgl="2011-08-21 08:00:00"
wasserstand="200" offen="0" zeitpunkt_vhr="2011-08-24 12:00:00"/>
</markers>
```

Abbildung 6-14 Aufruf der in PHP erstellten XML-Datei

4. Schritt Die Karte erstellen

Ist der Export nach XML erfolgreich, so kann die Google Maps-Karte erzeugt werden. Um eine Datei, wie die *xml_startseite.php* zu laden, muss eigens eine Funktion erzeugt werden, die das Laden der Funktion realisiert. Dazu wurde eine *function downloadUrl(url, callback)* definiert. *url* gibt den Dateinamen der Seite an, die die XML-Datei erzeugt, *callback* die Funktion, „die aufgerufen wird, wenn die XML-Ausgabe an JavaScript zurückgegeben wird.“ (Google Inc., PHP/MySQL mit Google Maps verwenden, 2011)

Nachdem die Funktion definiert wurde, kann sie nun mit der *xml_startseite.php* und einer Rückruffunktion aufgerufen werden. In der Rückruffunktion werden alle Marker-Elemente gesucht und durchlaufen. Für jedes Element im Marker werden Name, Typ, Station, Nummer, Zeitpunkt und Wasserstand des Pegels, Zeitpunkt und Status des Wehrs sowie dessen Koordinaten abgerufen. Ein neuer Marker wird mit den abgerufenen Koordinaten erstellt, und ein Infofenster an diesen gehangen, das mit den Informationen oben genannter Attribute versehen wird, wenn der Nutzer auf den Marker der Karte klickt.

6.7 Die Visualisierung von Überflutungsflächen

Das Hochwasserinformationssystem ermöglicht das Fluten der Polderflächen zu visualisieren. Wie im vorherigen Kapitel gezeigt geschieht dies durch KML-Dateien, die aus Rasterbildern errechnet wurden. Das sollte ausschließlich als Simulation bezeichnet werden, da die Visualisierung der Flächen mit entsprechender Unterscheidung des Wasserstands nur aufgrund des eingegebenen Pegelwerts in Bezug auf die im Raster angegebenen Höhenwerte berechnet wurde. Es besteht die Möglichkeit, die Pegelstände auf zwei Arten zu simulieren, die im Folgenden gezeigt werden.

6.7.1 Manuelle Pegelstandseingabe



Abbildung 6-15 Visualisierung manueller Pegelstandseingaben

Der Nutzer kann die Pegelstände anhand der Dropdown-Liste auf der rechten Seite verändern, daraufhin wird die Google Maps-Karte mit den entsprechenden KMZ-Dateien als Overlay neu geladen (siehe Abb. 6-15). Dazu wurde eine JavaScript-Datei (*polder.js*) eingebunden, und die darin definierten Funktionen *Garbe(Zahl)* oder *Wrechow(Zahl)* mit dem entsprechenden Pegelwert aufgerufen. Wie im vorherigen Kapitel beschrieben, wurden die KMZ-Dateien für bestimmte Wertebereiche (in 25 cm Abständen) erstellt. Wird der eingegebene Pegelwert in der Abfrage gefunden, wird ihm ein fester Wert zugeordnet, und von einem Startwert (im Polder Garbe 1650 cm) aus, mithilfe einer Schleife, in 25er-Schritten eine KMZ-Datei dazu geladen, bis der fest zugeordnete Wert erreicht wird. Die KMZ-Dateien werden partiell geladen. Das soll mit dem folgenden Schema (siehe Abb. 6-16) besser dargestellt werden:

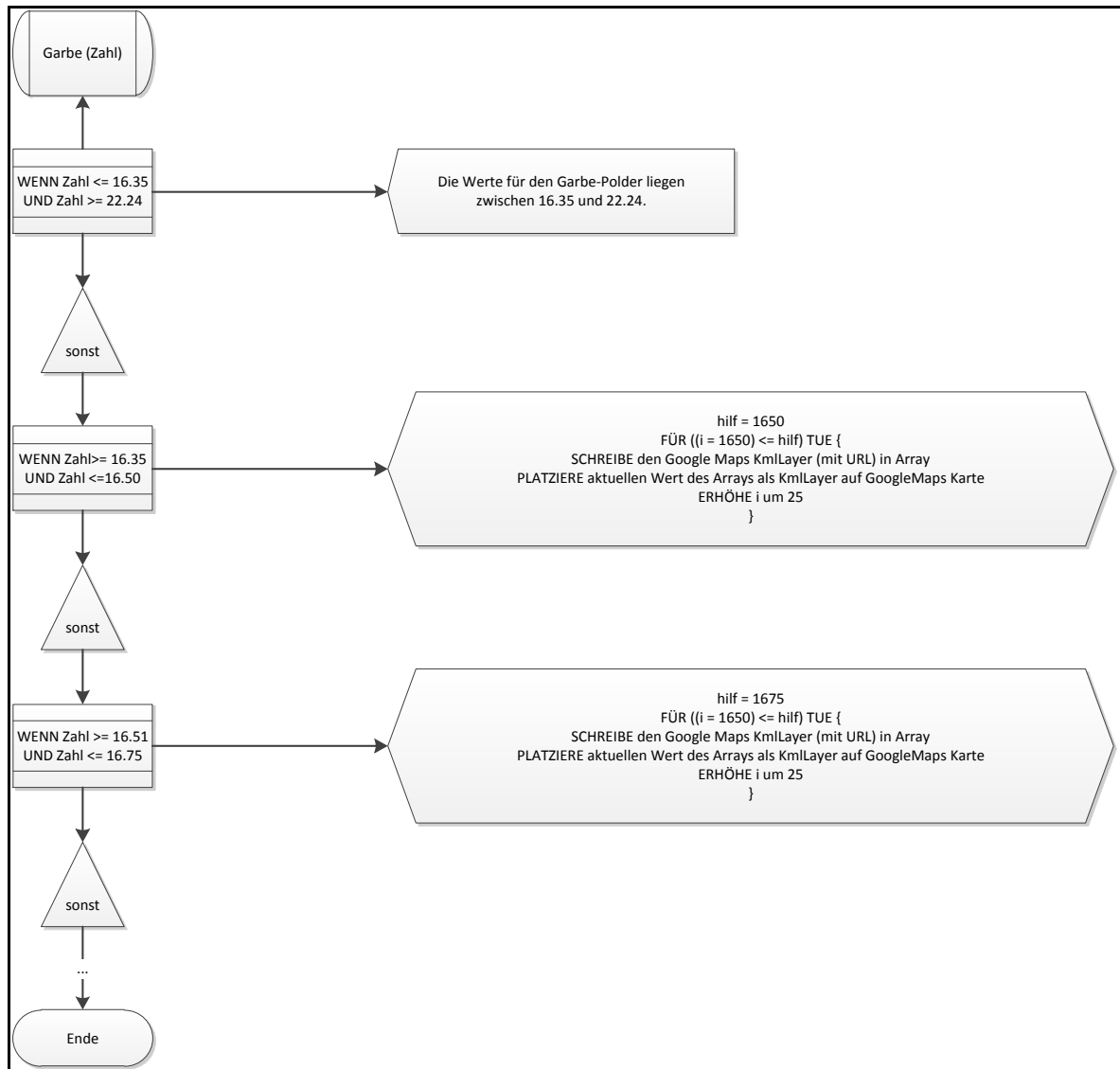


Abbildung 6-16 Schema des Funktionsaufrufs für das Visualisieren der Überschwemmungsflächen

Um die KMZ-Datei einzubinden, wird das KmlLayer-Objekt, für dessen Konstruktor die URL einer öffentlich verfügbaren KMZ-Datei benötigt wird, verwendet.

6.7.2 Visualisierung durch Datenbank-Pegelwerte

Bei der Visualisierung anhand des aktuellen Pegels in der Datenbank, wird dieser beim Laden der *index.php* ausgelesen und bei Klick auf den Button, an dieselbe JavaScript-Funktion übergeben.

7 ZUSAMMENFASSENDE DARSTELLUNG UND AUSBLICK

Ziel der Arbeit war es ein Hochwasserinformationssystem, mit Visualisierung der Bauwerke, sowie deren Daten und die Visualisierung der überschwemmten Flächen in den Poldern, zu entwerfen und zu entwickeln. Die Grundlagen wurden definiert, der Systementwurf und die Implementierung dokumentiert.

7.1 Zusammenfassung der Ergebnisse

Das Ergebnis des Entwurfs stellt einen Prototyp eines Hochwasserinformationssystems dar. Der Administrator kann im geschützten Bereich über einen Login neue Bauwerke zum System hinzufügen, sowie dessen Daten aktualisieren oder löschen. Die Bauwerke werden auf einer Google Maps Karte, abhängig von dessen Typ, mit einem entsprechenden Symbol, dargestellt und bei Klick auf den Marker mit Zusatzinformationen versehen. Die Visualisierung der Überschwemmungsflächen in den Poldern erfolgte durch Transformation der Rasterdaten in das Google KML-Format, die bei Anfrage als Overlay auf die Karte gelegt werden. Bauwerke können einzeln angezeigt werden, mit einer Tabelle, die alle Informationen auflistet, sowie eine Tabelle mit allen zum Bauwerk gehörenden Werten.

Der Prototyp hat ein gutes Anfangsstadium erreicht, bietet jedoch eine Reihe von Erweiterungsmöglichkeiten und Optimierungen, die noch umgesetzt werden können.

7.2 Ausblick

Eine Möglichkeit das System zu erweitern ist die Ausgabefunktion, die den Export der gewählten Bauwerksinformationen und –daten ins PDF-Format ermöglichen sollte. Zu prüfen wäre, ob der PDF-Export der Google Maps Karte ebenso möglich ist. Auch ein Export der Daten aller Bauwerke ins CSV-Format könnte zum Funktionsumfang gehören. Wünschenswert – bei einzelnen Bauwerken einen Zeitrahmen eingeben zu können, um Werte eines bestimmten Zeitintervalls anzeigen zu lassen. Ein weiterer Punkt der untersucht und umgesetzt werden müsste, ist die Langzeitspeicherung der Daten zur Beweissicherung der Bedienhandlungen bzw. Pegelstände an den wasserwirtschaftlichen Bauwerken, z.B. durch Export der Daten in das CSV-Format. Außerdem wäre eine Visualisie-

zung der Pegelwerte durch einen Graphen anstrebenswert – PHP bietet dazu vordefinierte Grafikfunktionen. Eine Verbesserung bedarf es ebenso bei der Darstellung der Überschwemmungsflächen in den Poldern. Hier gerät Google Maps bei der KML-Darstellung zu großer Datenmengen schnell an seine Grenzen und die eigentlich ausgefüllten Flächen erscheinen nur noch lückenhaft, bzw. erst bei höherer Zoomstufe.

Ein großer Vorteil des Systems ist jedoch, dass es ohne größere Anpassungen auf andere Flussbereiche Deutschlands erweitert werden kann. Ebenso kann die Datenbank mit weiteren Attributen versehen werden, um weitere relevante Daten hinzuzufügen, wie z.B. die Abflussmenge Q .

LITERATURVERZEICHNIS

(2009). *Wasserhaushaltsgesetz - WHG vom 31. Juli 2009 (BGBl. I S. 2585) das durch den Artikel 12 des Gesetzes vom 11. August 2010 (BGBl. I S. 1163) geändert worden ist.*

DATAKOM Buchverlag GmbH. *ITwissen.info*. Abgerufen am 02. 08 2011 von <http://www.itwissen.info/definition/lexikon/Mashup-mashup.html>

ESRI Deutschland. (2011). *ArcView*. Abgerufen am 03. 08 2011 von <http://esri.de/products/arcgis/arcview/index.html>

Google Inc. (2011). *Google Maps*. Abgerufen am 26. 07 2011 von <http://maps.google.de/maps?hl=de&ll=52.965184,11.863174&spn=0.256412,0.816422&z=11&vpsrc=6>

Google Inc. (2011). *Google Maps JavaScript API V3 – Allgemeines*. Abgerufen am 10. 08 2011 von <http://code.google.com/intl/de/apis/maps/documentation/javascript/basics.html>

Google Inc. (2011). *Google Maps-/Google Earth-APIs – Nutzungsbedingungen*. Abgerufen am 02. 08 2011 von <http://code.google.com/intl/de/apis/maps/terms.html>

Google Inc. (2011). *PHP/MySQL mit Google Maps verwenden*. Abgerufen am 10. 07 2011 von http://code.google.com/intl/de/apis/maps/articles/phpsqlajax_v3.html

Gunkel, C. (2010). *Google Earth: Möglichkeiten und Grenzen der Nutzung für geographische Anwendungen [Elektronische Ressource]*. München: GRIN Verlag GmbH.

Jarosch, H. (2010). *Grundkurs Datenbankentwurf : eine beispielorientierte Einführung für Studenten und Praktiker*. Wiesbaden: Vieweg + Teubner.

Kofler, M. (2005). *MySQL 5 : Einführung, Programmierung, Referenz*. München ; Boston [u.a.]: Addison-Wesley.

- Landesbetrieb für Hochwasserschutz und Wasserwirtschaft. (2011). *Hochwasservorhersagezentrale (HVZ)*. Abgerufen am 29. 07 2011 von <http://www.hochwasservorhersage.sachsen-anhalt.de/index.htm>
- NetBeans. (2011). *Home*. Abgerufen am 03. 08 2011 von http://netbeans.org/index_de.html
- Ormsby, T. (2004). *Getting to know ArcGIS desktop : basics of ArcView, ArcEditor, and ArcInfo* (2. Ausg.). Redlands, California: Environmental Systems Research Institute Inc., U.S.
- Patt, H. (2001). *Hochwasser-Handbuch: Auswirkungen und Schutz*. Berlin Heidelberg: Springer Verlag.
- Patt, H., & Gonsowski, P. (2011). *Wasserbau : Grundlagen, Gestaltung von wasserbaulichen Bauwerken und Anlagen*. Berlin ; Heidelberg: Springer .
- Priebe, D.-I. J. (2009). *Hochwasserschutzplan Aland - Ermittlung der Grundlagen zur Festlegung der Steuerrichtlinien der Bauwerke am Aland*. Planungsgesellschaft für Wasserbau & Wasserwirtschaft mbH - PROWA Neuruppin.
- ProgrammableWeb. (2010). *API Dashboard*. Abgerufen am 02. 08 2011 von <http://www.programmableweb.com/apis>
- Staatsbetrieb Geobasisinformation und Vermessung Sachsen (GeoSN). (kein Datum). *Geobasisinformation und Vermessung*. Abgerufen am 15. 08 2011 von <http://www.landesvermessung.sachsen.de/inhalt/etrs/grund/grund.html>
- Strobl, T., & Zunic, F. (2006). *Wasserbau. Aktuelle Grundlagen- Neue Entwicklungen*. Springer, Berlin.
- The PHP Group. (2011). *Was ist PHP?* Abgerufen am 04. 08 2011 von <http://www.php.net/manual/de/intro-what-is.php>

ABBILDUNGSVERZEICHNIS

Abbildung 2-1 Projektgebiet (Google Maps, 2011).....	7
Abbildung 2-2 Kartenansicht Hochwasservorhersagezentrale (Landesbetrieb für Hochwasserschutz und Wasserwirtschaft, 2011).....	8
Abbildung 2-3 Listenansicht Hochwasservorhersagezentrale (Landesbetrieb für Hochwasserschutz und Wasserwirtschaft, 2011).....	9
Abbildung 2-4 Querschnitt durch <i>a</i> ein festes Wehr und <i>b</i> ein bewegliches Wehr (Patt & Gonsowski, 2011, S. 122)	10
Abbildung 2-5 Die beliebtesten APIs als Mashups (ProgrammableWeb, 2010)...	12
Abbildung 3-1 Ausschnitt aus ArcMap.....	15
Abbildung 3-2 Ausschnitt aus ArcCatalog	15
Abbildung 3-3 Ausschnitt aus der NetBeans IDE 7.0	16
Abbildung 3-4 Darstellung der PHP-Funktionsweise	17
Abbildung 4-1 Benutzerrollenverteilung mit entsprechenden Aktionsmöglichkeiten	20
Abbildung 4-2 Unnormalisiertes Datenmodell für einen Pegel	21
Abbildung 4-3 1. Normalform.....	22
Abbildung 4-4 Tabelle bauwerk	23
Abbildung 4-5 Tabelle pegel.....	23
Abbildung 4-6 Tabelle wehr.....	24
Abbildung 4-7 Tabelle gewaesser	24
Abbildung 4-8 Entity-Relationship-Modell.....	25
Abbildung 5-1 Schema Raster zu KML am Beispiel Polder Wrechow.....	27
Abbildung 5-2 ASCII Grid-Datei.....	28
Abbildung 5-3 Ansicht der Rasterstreifen (Blöcke) mit Shape des Wrechow-Polders (Ausschnitt ArcMap).....	29
Abbildung 6-1 Struktur des Web-Interfaces.....	34
Abbildung 6-2 Schematischer Ablauf der wichtigsten Aktionen im Web-Interface	35
Abbildung 6-3 Layout-Entwurf der Seite <i>Zeige Liste aller Bauwerke</i>	36
Abbildung 6-4 Login-Formular	37
Abbildung 6-5 Geschützter-Bereich.....	38
Abbildung 6-6 Anlegen eines Bauwerks	38

Abbildung 6-7 Bauwerksdaten einlesen / eingeben.....	39
Abbildung 6-8 Pegelwert manuell eingeben	39
Abbildung 6-9 Formular Pegelwerte aus Excel einlesen	40
Abbildung 6-10 Exportierte Excel-Tabelle	40
Abbildung 6-11 Aktion am Wehr durchführen.....	41
Abbildung 6-12 Datenbank bereinigen	41
Abbildung 6-13 Aufruf der in PHP erstellten XML-Datei	42
Abbildung 6-14 Visualisierung manueller Pegelstandseingaben	44
Abbildung 6-15 Schema des Funktionsaufrufs für das Visualisieren der Überschwemmungsflächen	45
Abbildung A-0-1 Ansicht einer einfachen KML-Datei die eine Ortsmarke enthält.	53

TABELLENVERZEICHNIS

Tabelle 1 Wertebereich und Rundungswerte auf 25	31
---	----

ANHANG

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Placemark>
    <name>Simple placemark</name>
    <description>Attached to the ground. Intelligently places itself
      at the height of the underlying terrain.</description>
    <Point>
      <coordinates>-122.0822035425683,37.42228990140251,0</coordinates>
    </Point>
  </Placemark>
</kml>
```

Abbildung A-0-1 Ansicht einer einfachen KML-Datei die eine Ortsmarke enthält

EIDESSTATTLICHE ERKLÄRUNG

Hiermit versichere ich, Fabian Schulte, die vorliegende Bachelorarbeit „Entwicklung eines Hochwasserinformationssystems – Am Beispiel der Aland-Mündung“ ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Neuruppin, den 09.September 2011

Fabian Schulte

DANKSAGUNG

Hiermit möchte ich mich bei allen bedanken, die mich bei der Erstellung der Bachelorarbeit unterstützt haben.

Besonderer Dank gilt dem Team der Planungsgesellschaft für Wasserbau und Wasserwirtschaft mbH, PROWA Neuruppin, das mir die Möglichkeit gegeben hat, die Arbeit in diesem Unternehmen zu schreiben und mich in vielerlei Hinsicht unterstützt hat.

Ein großes Dankeschön gilt auch Herrn Dipl.-Ing. oec. Hans-Jörg Steingraf, für die Zusammenarbeit und Unterstützung

Neuruppin, September 2011