

Mobiharz - schnurlos durch den Harz
PeNG – Prüfungsergebnisse Next Generation

Projektbericht 04.00 – 02.01

Betreuer:

Prof. Dr. B. Müller
Prof. Dr. H. Eirund

Studierende:

Georges Gebara
Nico Hoffmann
Alexander Schirge
Edgar Soldin

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ein denkbares Anwendungsszenario	1
1.2	Inhaltliche Aspekte	2
1.3	Technische Aspekte	2
1.4	Umsetzung	3
2	Was ist WAP	3
2.1	WAP - Wireless Application Protokoll - das Protokoll für schnurlose Anwendungen	4
2.2	Schichtenarchitektur	5
2.3	Verbindung	6
2.4	Fazit	6
3	Grundlagen, Aufbau von WML	7
4	PeNG;-) (Prüfungsergebnisse Next Generation)	10
4.1	Aufgabe	10
4.2	Spezielle Anforderungen der Anwendung	10
4.3	Hardwarearchitektur	11
4.4	Softwarearchitektur	12
4.5	Navigationsstruktur	14
4.6	Übersicht Navigation	15
5	Ausblick	16
6	Anhang	17
6.1	Benutzung von PeNG	17
6.2	Installation von PeNG	18
6.3	Marktanalyse Mobiharz (Stand: 17.05.00)	20
6.4	Klassendiagramm mit Ein-/Ausgabevisualisierung	25

1 Einleitung

Ziel des Projekts Mobiharz ist die Konzeption und Erstellung einer Internet- und Wap-basierten Anwendung zur Unterstützung der Reiseplanung und -organisation in der Harz-Region.

Die geplante Durchführung erfolgt durch die Hochschule Harz unter der Leitung von Prof. Dr. Helmut Eirund und Prof. Dr. Bernd Müller sowie regionale und überregionale Partner. Als Durchführungszeitraum sind 3 Jahre vorgesehen.

Die im folgenden beschriebenen Ergebnisse und Konzepte entstanden unter der Mitarbeit von Studenten des Studiengangs Medieninformatik der Hochschule Harz (Nico Hoffmann, Georges Gebara, Edgar Soldin, Alexander Schirge).

1.1 *Ein denkbare Anwendungsszenario*

Ein potentieller Reisender interessiert sich für den Harz und erkundigt sich am PC zu Hause via Internet über die Region Harz, deren touristische Angebote sowie im geplanten Reisezeitraum stattfindende Veranstaltungen. Aufgrund der positiven Wettervorhersage für Region und Zeitraum beginnt er mit der konkreten Reiseplanung. Mithilfe der Abfrage möglicher Verkehrsmittel und deren Fahrpläne fällt die Entscheidung für eine An-/Abreise per Bahn sowie die Nutzung des Personennahverkehrs (PNV) vor Ort. Nach der Bestellung der entsprechenden Tickets folgt die Auswahl und Buchung der Unterkunft. Abschliessend reserviert er noch die Eintrittskarten eines stattfindenden Konzertes und lässt sich eine Liste von empfohlenen touristischen Aktivitäten erstellen.

Vor Ort unterwegs erkundigt sich der Reisende mit seinem mobilen Endgerät (WAP-Handy) über das beeindruckende Gebäude vor dem er gerade steht und erfährt, dass hier auch Theatervorführungen stattfinden. Kurz entschlossen bucht er für die Vorführung am nächsten Abend und folgt weiter dem vorgeschlagenen Stadtrundgang. Am nächsten Tag im Wald wird er von enormen Durst befallen und erkundigt sich via WAP über die gastronomischen Einrichtungen in der Nähe und folgt der Wegbeschreibung zum nächsten geöffneten Lokal. Der folgenden Wanderunlust begegnet der Reisende indem er sich online über in der Nähe liegende Bushaltestellen und den nächsten erreichbaren Bus

informiert. Die erhaltene Wegbeschreibung mit dem Hinweis, dass noch genug Zeit zur Verfügung steht, lässt ihn hochzufrieden in Richtung Bus schlendern.

Zur Realisierung dieses Systems sind folgende inhaltliche und technische Aspekte relevant.

1.2 Inhaltliche Aspekte

Die Bereitstellung der relevanten Informationen erfolgt über die entsprechenden Projektpartner (PNV, Touristeninformation, usw.). Die inhaltliche Verantwortung verbleibt somit beim ursprünglichen Anbieter. Struktur und Umfang der Daten werden durch das System entsprechend der benutzten Endgeräte angepasst.

Mögliche Daten zur Reiseplanung und Tourismus- und Service-Angeboten:

- Fahrpläne des PNV und anderer Beförderungsorganisationen
- Touristische Angebote/Attraktionen
- Wanderrouten
- Wetter
- ...

1.3 Technische Aspekte

- Identifizierung verschiedener Endgeräte und Browsertypen für die entsprechende Ausgabe (WML/HTML) und Anpassung der Struktur und des Umfangs

Endgeräte	Browser
Personal Computer (PC)	Web-Browser
Personal Digital Assistant (PDA)	WAP-/Web-Browser
Mobiltelefon	WAP-Browser

- Personalisierung (Benutzerprofil, Login, Endgerätecode - IMEI)
- Lokalisierung (Standortinformationen der Provider, manuelle Eingabe von Standortinformationen)
- Einbindung der (heterogenen) Datenquellen

1.4 Umsetzung

Die für die Realisierung notwendigen Grundkenntnisse sollten durch Konzeption und Umsetzung einer Beispielanwendung gewonnen werden. Die Wahl fiel auf die Abfrage der Prüfungsergebnisse der Studenten, da diese viele Gemeinsamkeiten mit der Projektaufgabenstellung aufweist (Personalisierung, Datenbankzugriff, Formatierung der Ausgabe).

Das Ergebnis dieser Bemühungen ist PeNG;-) (Prüfungsergebnisse Next Generation) und wird ab Abschnitt 3 dieses Dokumentes näher erläutert.

2 Was ist WAP

Die Idee, Daten über vorhandene oder noch zu entwickelnde Mobilfunknetze zu übertragen, ist Wirklichkeit geworden. Netzinhalte und -dienste sind für mobile Endgeräte wie z.B. Handys, Pager oder Palmtops erreichbar. Alle diese Geräte sind gekennzeichnet durch kleine Bildschirme, eingeschränkte Tastaturen, geringe Bandbreite der Verbindung und wenig Arbeitsspeicher. Diese Punkte gilt es zu beachten, wenn über "mit dem Handy im Internet zu surfen" gesprochen wird.

Aufgrund dieser "Einschränkungen" ist es schwierig, oft sogar unmöglich, vorhandene Internetinhalte (HTML, Grafiken, Multimedia) darzustellen oder bedienbar zu halten. Desweiteren liegt die aktuelle Bandbreite von Mobilfunkgeräten bei maximal 9400 Bit pro Sekunde und dies nur wenn der Mobilfunkempfang völlig ungestört ist.

Es ist also ersichtlich, daß "Internetsurfen" illusorisch und allenfalls eingeschränkt unter Minimierung der zu übertragenden Datenmenge und darzustellenden Information möglich ist.

Die Entwicklung der von Standards, die diese Punkte berücksichtigten, begann 1997.

Nachdem zur Cebit einige proprietäre Protokolle einzelner Hersteller vorgestellt wurden und absehbar war, daß keiner dieser Standards weit genug entwickelt war, um sich auf dem Markt und auch bei der Konkurrenz durchzusetzen, entschlossen sich einige dieser Hersteller ein Bündnis (Joint-Venture) einzugehen. Im Juni 97 wurde das WAP-Forum gegründet.

Ziel ist es einen offenen, plattform- und datenkanalunabhängigen Standard zu schaffen, der die Grundlage für Datenübertragung an mobile Geräte schaffen soll. Um dies zu gewährleisten ist die Mitgliedschaft im WAP-Forum offen für jeden Industriepartner (Hersteller von Handys, Mobilfunknetzbetreiber etc.).

2.1 WAP - Wireless Application Protokoll - das Protokoll für schnurlose Anwendungen

Schon im April 1997 wurden die ersten Spezifikationen zu WAP 1.0 veröffentlicht, die im September 1999 durch Version 1.1 zu großen Teilen ersetzt wurde. Die Neuspezifizierung führte zu einer Inkompatibilität zwischen beiden Versionen.

Davon betroffen sind u.a. frühe Chargen des Mobiltelefones Siemens S25, die WAP 1.0 implementierten.

Erst mit WAP 1.1 und der weltweiten Herstellung von Mobilfunkgeräten, die diesen Standard unterstützten begann WAP sich auf dem Markt durchzusetzen. Inzwischen (Januar 2000) ist Version 1.2 verabschiedet.

WAP ist ein völlig neues Konzept. Es bietet datenorientierte Dienste für den Massenmarkt und kann durch die drahtlose Vernetzung weit mehr Nutzer als durch herkömmliche, kabelabhängige Technologien erreichen. Es existiert als Spezifikation, ähnlich HTML, und

überlässt die konkrete Implementierung den Geräteherstellern, Internetanbietern und anderen Nutzern.

Spezifiziert sind unter anderem:

eine Auszeichnungssprache(WML),

eine Scriptsprache(WML Script),

ein Microbrowser(für die Darstellung der WML Seiten verantwortlich),

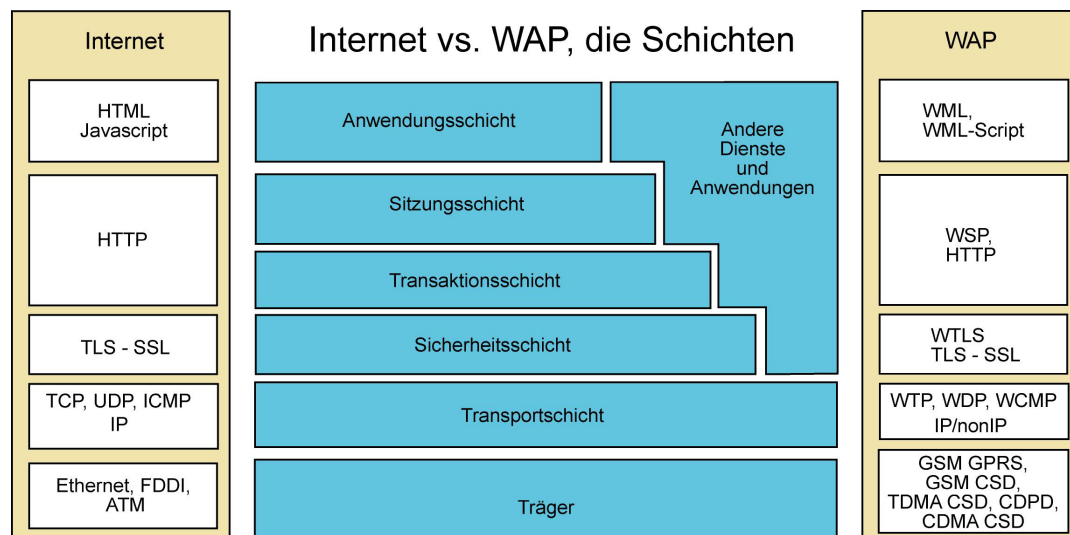
eine Telefonschnittstelle (WTAI - ermöglicht Telefonfunktionen aus WML Script aufzurufen),

neue Dateiformate(Visitenkarten,Kalendereinträge,Grafik-WBMP),

eine Schichtenarchitektur der Protokolle.

2.2 Schichtenarchitektur

WAP ist in Schichten definiert, die dem Internet nachempfunden wurden. Dies führt zu einer starken Unabhängigkeit der einzelnen Schichten untereinander, ermöglicht vorhandene Technologien einzubinden oder einzelne Schichten direkt anzusprechen.



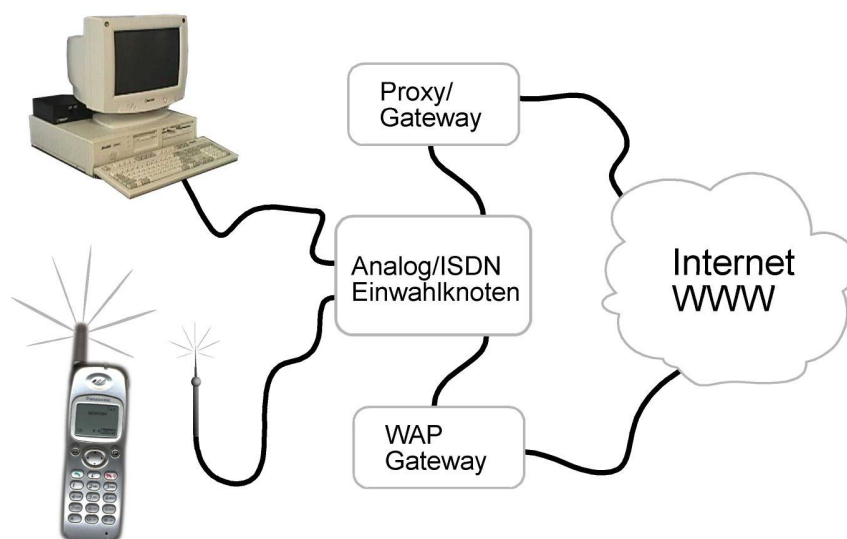
Vergleich bestehender Protokolle des Internets und WAP und Zuordnung zu Schichten

Aus der vorliegenden Grafik wird ersichtlich, daß WAP durchaus ermöglicht vorhandene Protokolle zu nutzen und eigene Protokolle auf diese aufzusetzen. Bei einem Vergleich mit der herkömmlichen DFÜ-Verbindung wird es noch deutlicher.

2.3 Verbindung

Die herkömmliche Einwahl bei einem Internetzugangsanbieter ist prinzipiell nichts anderes als die Einwahl mit dem mobilen Gerät (DFÜ-Verbindung). Es wird eine Zugangstelefonnummer gewählt und diese stellt eine Verbindung zum Internet her.

Augenfälliger Unterschied ist ein zwischengeschaltetes WAP-Gateway, welches die zu empfangenden Daten komprimieren und zu sendende dekomprimieren muß, um die Bandbreite zum Mobilgerät optimal nutzen zu können. Das Gateway ist vergleichbar mit dem Prinzip eines Proxy- oder Gatewayservers in der herkömmlichen DFÜ-Verbindung.



Vergleich von DFÜ und WAP Verbindung

2.4 Fazit

WAP ist eine faszinierende neue Technologie, deren einziger Vorteil die Schnurlosigkeit ist (momentan). Endgeräte kranken an mangelhafter Hardware (Bildschirm, Tastatur) und langsamer Verbindungsgeschwindigkeit. Dieser Fakt ist um so wichtiger, da momentan in

allen Mobilfunknetzen 1 Minute „wappen“ 0,39 DM kostet. Abhilfe ist in Sicht und heißt HSCSD (57,6kBit), GPRS(115kBit) oder UMTS(2MBit). Achtung: Die Geschwindigkeitsangaben sind stark abhängig von der Verbindungsqualität und Konfiguration des Anbieters. So kann und wird UMTS bei wahrscheinlicher Einführung 2002 in den meisten Zellen auf eine wesentlich geringere Bandbreite begrenzt werden, um diesen Service möglichst vielen zahlenden Nutzern zur Verfügung zu stellen. Günstiger wird UMTS deswegen kaum, da für die Lizenzen und Aufrüstung der Mobilfunknetze Milliarden investiert wurden.

GPRS wird gerade eingeführt und hat den Vorteil verbindungslos zu sein, das heisst wird nach Datenmenge abgerechnet. Die Preise hierfür liegen zwischen 0,19 und 0,69 DM pro angefangene 10 kByte. Es bleibt abzuwarten ob GPRS somit günstiger ist.

Letztendlich sind die schnurlose Verfügbarkeit von Datendiensten und standardisierte Schnittstellen die zukunftsweisenden Merkmale von Wap und darauf basierende Geräte, wie auch Applikationen werden die Informationsaustausch der Zukunft vom Kabel trennen.

3 Grundlagen, Aufbau von WML

Ein WML-Dokument ist ähnlich einem HTML-Dokument durch verschiedene Tags aufgegliedert, welche den in ihnen stehenden Inhalt definieren, kann aber im Gegensatz zu diesen mehrere nacheinander folgende Seiten enthalten. Dies dient einer für den Nutzer komfortablen Informationsaufbereitung, da aufgrund der geringen Bildschirmauflösung der Endgeräte die Informationen auf einzelne Seiten zu verteilen sind, um umständliches und unübersichtliches Scrollen zu umgehen.

Die Informationen eines einzelnen Seiten sind innerhalb einer sogenannten "Karte" definiert, welche im WML-Code durch die Tags `<card>` und `</card>` abgegrenzt wird. Karten sind die eigentlichen Informationsträger eines WML-Dokumentes.

Mehrere dieser Karten werden zu einem Stapel, dem sogenannten "Deck", zusammengefasst. Das Deck ist durch die Tags `<wml>` und `</wml>` definiert und umfasst den gesamten Inhalt eines WML-Dokumentes. Hier können zu den

einzelnen enthaltenen Karten auch Voreinstellungen definiert werden, was der geringen Übertragungsbandbreite des Mediums entgegenkommt.

Die einzelnen Karten und Decks sollten im Regelfall untereinander verlinkt sein, so daß sie bequem angesprungen werden können. Hierzu gibt es als Navigationselement wie in HTML die Links.

Zur Navigation sind weiterhin Möglichkeiten des Users, Daten einzugeben, entscheidend. Hier stehen wie in HTML definierbare alphanumerische Eingabefelder, Checkboxes zur Mehrfachauswahl und Radiobuttons (zur Auswahl eines Wertes aus einer Wertemenge) zur Verfügung.

Die Navigation erfolgt unter Zuhilfenahme der Cursor- und Funktionstasten, ist also endgerätespezifisch und daher unterschiedlich leicht oder schwer durchschaubar.

Ein weiteres Problem ist die Darstellung, da bei manchen Endgeräten weder Kartentitel noch Namen von Eingabefeldern ausgegeben werden und sich die Links mehr oder weniger leicht zugänglich unter einer Optionstaste verstecken.

Allgemeine Orientierungshilfen kann man über die Textformatierung realisieren; hier stehen Veränderungen der Schriftgrösse, der Dicke, Kursivsetzung und Unterstreichungen der gerätespezifischen Schriftart zur Verfügung. Als weitere Gestaltungsmöglichkeiten kommen noch Tabellen und 1-bit-Grafiken hinzu.

3.1.1 Übersicht über ein WML-Dokument

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
    "http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
  <template>
    Für alle Karten geltende Voreinstellungen
    Link im Optionen-Menü; entspricht History.Back()
    <do type="prev" label="Zurueck" name="Zurueck"><prev/></do>
  </template>

  <card id="Karte_1">
    Link im Optionen-Menü; Absenden der Eingaben
    <do type="accept" label="Absenden" name="Absenden">
      <go href="http://193.175.32.57/mobiserv/Wrapper">
        <postfield name="Semester" value="$(Semester)"/>
        <postfield name="Fach" value="$(Fach)"/>
        <postfield name="Status" value="$(Status)"/>
        <postfield name="Session" value="-123456"/>
      </go>
    </do>

    <p>Semester:
      Checkboxes
      <select name="Semester" title="Semester" multiple="true">
        <option value="WS97">"WS97"</option>
        <option value="SS98">"SS98"</option>
        <option value="WS98">"WS98"</option>
      </select>
    </p>
    <p>Status:
      Radio-Buttons
      <select name="Status" title="Status">
        <option value="be">bestanden</option>
        <option value="nb">nicht best.</option>
        <option value="an">angemeldet</option>
      </select>
    </p>
    <p>Fach:
      Eingabefeld für exakt 5 Nummern
      <input name="Fach" title="Fach" type="text" format="5N"/>
    </p>
    <p>
      Link auf der Seite; Absenden der Eingaben
      <a href="http://193.175.32.57/mobiserv/Wrapper?Semester=$(Semester) &
        Fach=$(Fach) & Status=$(Status) & Session=-123456"
        title="Absenden">Absenden</a>
    </p>
    <p>
      Link auf der Seite; referenziert Karte innerhalb des Decks
      <a href="#Uebersicht" title="Zurueck">Zurueck</a>
    </p>
  </card>
  <card id="Uebersicht">
    Karteninhalt (Text, Grafiken, Eingabefelder, Links)
  </card>
</wml>
```

4 PeNG;-) (Prüfungsergebnisse Next Generation)

4.1 Aufgabe

Die Aufgabe zum konkreten WAP-Projekt entstand aufgrund eines Vorschlags von Herrn Prof.Dr.B.Müller und Herrn Prof.Dr.H.Eirund. Diese Idee wurde dann im weiteren noch verfeinert.

Letztendlich entstand ein Service, der es ermöglicht, via WAP Prüfungsergebnisse abzurufen. Dieser Dienst kann von allen Studenten der Hochschule Harz in Anspruch genommen werden, ein WAP-Handy vorausgesetzt. Bei der Implementierung spielt lediglich der WAP 1.1-Standard eine Rolle, dass heißt mit älteren WAP-Handies (z.B. Siemens S25) kann der Service nicht genutzt werden. Das Ausprobieren und Vertrautmachen mit der WAP-Technik steht im Vordergrund. Auch soll eine Grundlage für das spätere Projekt MobiHarz geschaffen werden. Die Schichtenarchitektur der Anwendung kann also später in ähnlicher Form angewandt werden.

4.2 Spezielle Anforderungen der Anwendung

So erfordern die sensiblen Daten über Prüfungen und deren Ergebnisse spezielle Sicherungsmechanismen. So muss der Nutzer sich anmelden, bevor er auf seine Daten zugreifen kann. Dies erfolgt über die an der Hochschule übliche Authentifizierung (U-Nummer/Passwort).

Desweiteren ist ein Sitzungs-Management (Session-Management) implementiert, das mehrere Zugriffe nach einmaliger Anmeldung ermöglicht.

Nach dem Anmelden werden alle in der Datenbank vermerkten Prüfungen des Nutzers im jeweils aktuellen Semester sortiert angezeigt. Eine Sortierung erfolgt nach bestanden, nicht bestanden, angemeldet und/oder einem Semester.

Der Nutzer hat die Möglichkeit zu einer erweiterten Suche, dass heißt, er kann selbst nach bestimmten Attributen suchen.

Ein wichtiger Punkt ist, die zu übertragenden Datenmengen möglichst klein zu halten. So werden Abkürzungen im Userinterface verwendet und Übersichten angeboten.

Die WML-Seiten werden möglichst textbasiert gestaltet, um die geringe Bandbreite nicht auch noch für Bilder zu nutzen. Auch die erweiterte Suche kann, richtig eingesetzt, helfen, die Datenmengen zu minimieren.

Die Nutzerführung ist dem Medium angemessen. So sind Links klar benannt und verwendete Abkürzungen leicht verständlich. Eingaben werden eingeschränkt, das heißt, möglichst wenig Texteingabe vom User erwartet. Dazu dienen Pulldownmenüs oder Schaltflächen, welche dynamisch erzeugt werden.

Elementar ist auch die strikte Einhaltung des WML 1.1 Syntax und der Verzicht auf eventuelle Sonderfunktionen bestimmter Hersteller die nicht generell unterstützt werden. Zum Testen dienen 2 gebräuchliche Handies (Siemens S35i, Motorola P7389e).

Natürlich werden auch Fehlermeldungen selbst für den technisch nicht versierten Nutzer verständlich ausgegeben.

Die Applikation soll all diesen Anforderungen gerecht werden, und muss in dem gesetzten Zeitrahmen arbeitsteilig realisiert werden.

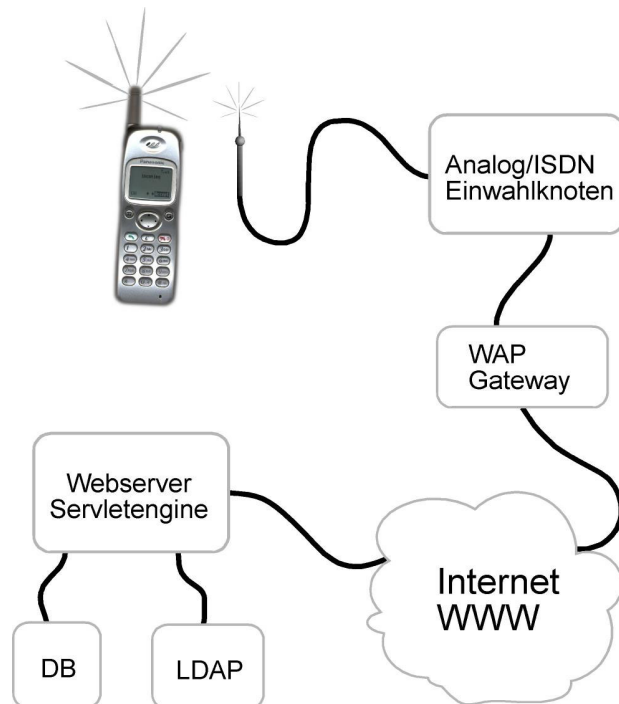
4.3 Hardwarearchitektur

Um WML erfolgreich zum Nutzer zu befördern bedarf es einiger Vorbereitung und technischer Vorraussetzungen (sh. 2.3).

Zum Einen muss eine Anbindung der mobilen Endgeräte an das Internet gewährleistet werden. Dazu gehören die Bereitstellung einer Einwahlnummer, eines DFÜ-Servers und eines WAP-Gateways. Dieser Part wird momentan als Service der Netzbetreiber angeboten. Es ist allerdings auch möglich und denkbar, um dem Nutzer Kosten zu ersparen, diesen Service eigenständig anzubieten.

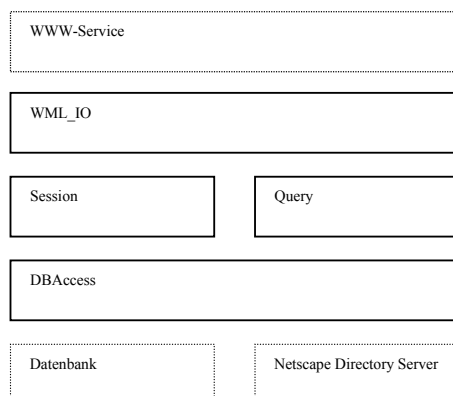
Zum Anderen bedarf es eines WWW-Servers, der in der Lage ist WAP-spezifische Contentformate zum Herunterladen bereitzuhalten. Wichtig für PeNG;-) ist eine Servleterweiterung des Servers, die es ermöglicht die in Java programmierte Anwendung in den Server zu integrieren.

Desweiteren setzt das Funktionieren der Anwendung einen funktionsfähigen LDAP-Dienst für die Authentifizierung und die Datenbank mit den Prüfungsergebnissen voraus.



Übersicht der Anwendung PeNG

4.4 Softwarearchitektur



Die Softwarearchitektur bildet die Grundlage zum arbeitsteiligen Programmieren der Applikation. Hierbei wurden die benötigten Klassen und Schichten mit Schnittstellen konzeptioniert.

Die Klassen lassen sich in ein gebräuchliches Schichtenmodell einordnen.

4.4.1 WML_IO

Die oberste Ebene der Anwendung stellt die Ein- und Ausgabeklasse WML_IO dar. Diese Klasse wird vom WWW-Server angesprochen und bekommt von dort die benötigten Parameter, etwa für eine erweiterte Suche.

Hier wird anhand der Parameter entschieden, ob ein Login oder ein nachfolgender Request vorliegt. Dementsprechend werden WML-Seiten erstellt und zurückgeliefert.

Examenslisten werden hier sortiert und entsprechend dem WML 1.1 Syntax in WML-Cards eingeordnet.

Ein elementarer Bestandteil ist die Zerteilung und anschließende Verlinkung von Ergebnismengen. Dadurch wird die maximale Datenmenge für ein Endgerät nicht überschritten.

Die zweite Schicht wird durch 2 Klassen repräsentiert: die Klasse Session und Query.

4.4.2 Session

Session ist zuständig für das Verwalten von Sitzungen, und ist deshalb auch statisch.

Attribute einer Sitzung werden hier gehalten und können durch entsprechende Methoden abgefragt werden.

Außerdem ist hier das Sitzungsmanagement realisiert. Eine Sitzung wird nach 5 Minuten ohne Aktivität gelöscht.

Im Falle eines Logins wird von hier aus die Authentifizierung durch ein DBAccess-Objekt angestoßen und bei Erfolg wird eine Datenbankverbindung erzeugt und dieser Sitzung in der Klasse Session zugeordnet.

4.4.3 Query

Des Weiteren befindet sich Query auf dieser Schicht. Query bekommt Parameter für eine Anfrage von WML_IO und leitet diese an DBAccess weiter, wo dann Examenslisten zusammengestellt werden. Die entsprechenden Ergebnismengen werden wieder an WML_IO zurückgegeben. Sitzungsspezifische Attribute (Matrikelnummer, Datenbankzugriffsobjekt,...) für eine Anfrage werden hier aus Session geholt.

4.4.4 DBAccess

Die Klasse DBAccess stellt die unterste Schicht der Architektur dar. Hier findet der eigentliche Datenbankzugriff statt. Examenslisten und/oder Semesterlisten werden entsprechend den Anfragen zusammengestellt. Auch werden hier Benutzer über einen Netscape Directory Server der Hochschule Harz (info.hs-harz.de) authentifiziert. Datenbankinhalte werden hier in eigene Typen verwandelt. Dazu dient eine Reihe von Hilfsklassen (z.B. Uid, Pwd, Sem, SemList, Exam, ExamList).

Ein ausführliches Klassendiagramm mit Ablaufplan liegt als Anlage bei (Anlage 1).

4.5 Navigationsstruktur

Die Navigation der Anwendung spielt sich auf drei Ebenen ab .

4.5.1 Loginebene

Während des Logins muß der User seine uNummer und sein Paßwort eingeben und abschicken; hat er einen Eingabefehler begangen, kommt eine entsprechende Fehlermeldung zurück ("User unbekannt" oder "Paßwort falsch") und er kann erneut versuchen, sich einzuloggen.

Die Eingabefelder akzeptieren nur exakt 5 Zahlen für die uNummer und 8 alphanumerische Zeichen für das Paßwort, um Eingabefehler von vornherein zu minimieren.

4.5.2 Übersichtsebene

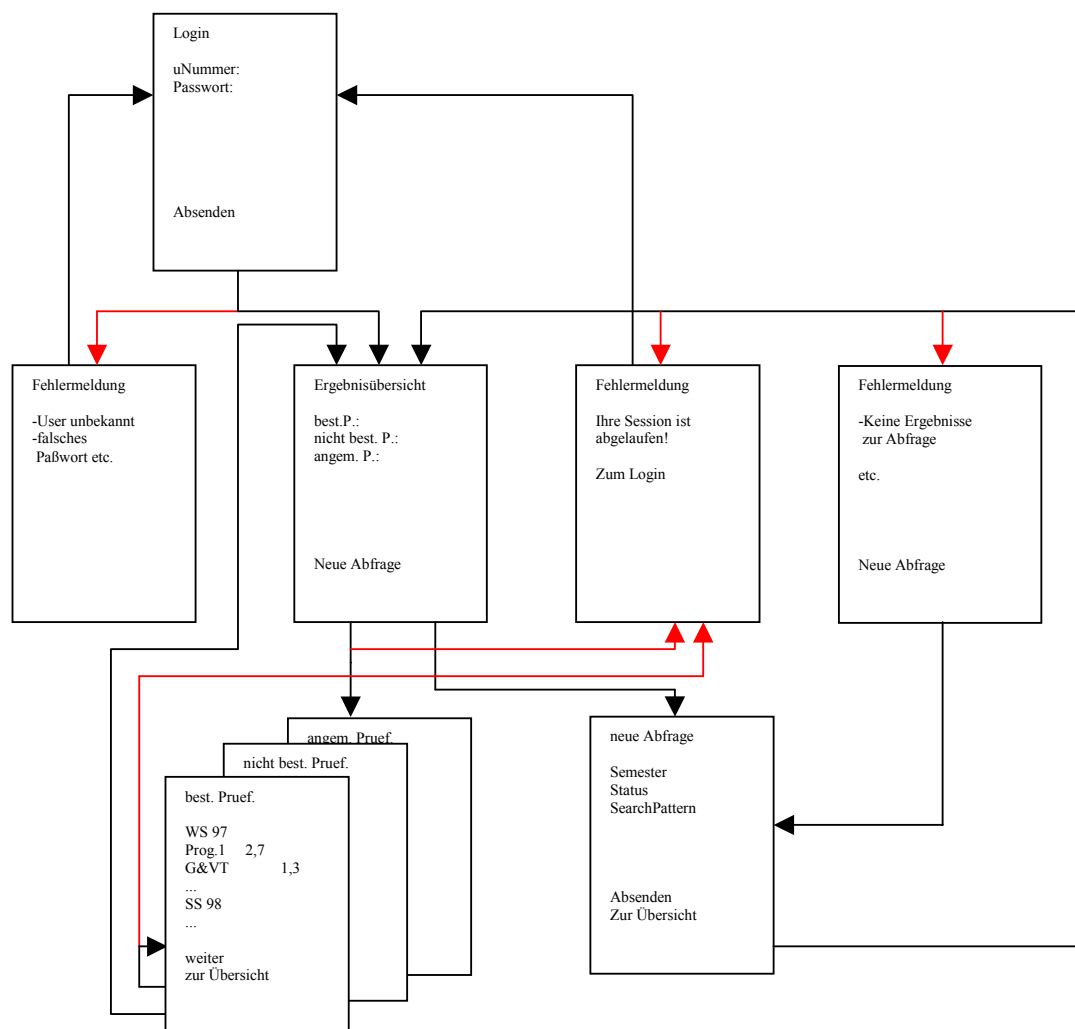
Hat der Nutzer sich korrekt angemeldet, verläßt er die Loginebene und erhält eine Übersicht der Anzahl der bestandenen, nicht bestandenen und angemeldeten Prüfungen des aktuellen Semesters mit Links zu den einzelnen Prüfungskategorien und einer Suchmaske für erneute Abfragen.

4.5.3 Ergebnisebene

Hier kann sich der Nutzer die Ergebnisse der einzelnen Kategorien ansehen oder eine differenziertere Abfrage stellen, die ihm eine neue Übersicht liefert. Falls keine Ergebnisse

zu der spezifischen Abfrage existieren, wird dies dem User in Form einer Fehlermeldung mitgeteilt und er kann eine neue Abfrage stellen. Falls die Session inzwischen abgelaufen sein sollte, erscheint eine Fehlermeldung und der Nutzer kann sich erneut einloggen. Um dem User eine Möglichkeit zu geben, umständliche Optionenmenüs zu vermeiden, sind alle Links auch direkt auf die einzelnen Karten geschrieben.

4.6 Übersicht Navigation



5 Ausblick

Die realisierte Anwendung PeNG;-) ist das erste aktive WAP-Angebot der Hochschule Harz. Sie bietet den Studenten eine neue, von PC/Internet unabhängige Möglichkeit, persönliche Daten abzurufen.

Im Hinblick auf die zu erwartende Verbreitung WAP-fähiger Endgeräte gewinnt dieses Angebot an Bedeutung und kann durch die Einrichtung eines WAP-Gateways an der Hochschule noch erweitert werden. Eine direkte WAP-Einwahl in das Hochschulnetz ermöglicht eine preiswerte Alternative zur Einwahl beim Mobilfunkanbieter.

Für das Projekt Mobiharz sind die gewonnenen Erkenntnisse von besonderer Bedeutung. Die realisierte Software- und Systemarchitektur kann nun getestet und beurteilt werden. Erfahrungen und erkannte Probleme können in die Umsetzung des Mobiharz Projektes einfließen.

Grundsätzlich kann gesagt werden, dass die Erfahrungen mit der WAP Technologie durchaus positiv sind. Wenn in Betracht gezogen wird, dass WAP sich noch in „den Kinderschuhen“ befindet, sollten viele der „Kinderkrankheiten“ wie:

Darstellungsunterschiede auf verschiedenen Endgeräten oder Deckgrößenbeschränkung von 1,4 kByte in Zukunft „geheilt“ werden.

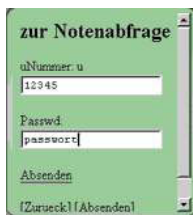
Desweiteren sollte davon ausgegangen werden, daß mit fortschreitender Verbreitung von WAP auch die Eingabe und Ausgabeschnittstellen verbessert und erweitert werden.

6 Anhang

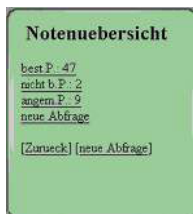
6.1 Benutzung von PeNG



Von wap.hs-harz.de kommt der Nutzer über den Link Prüfungsergebnisse zur Anmeldung.



Dort sind die u-Nummer **ohne das "u"** und das Passwort einzugeben.



Die Rückgabe ist eine Notenübersicht des aktuellen Semesters. Über die Links auf der Karte "Notenübersicht" ...



... kann eine detaillierte Auflistung der bestandenen, nicht bestandenen und angemeldeten Prüfungen angesehen ...



... oder eine neue Abfrage formuliert werden, welche wiederum in einer Notenübersicht resultiert.

6.2 Installation von PeNG

Die serverseitige javabasierte Anwendung besteht aus folgenden Dateien.

wap.jar	enthält die Java Klassen der Anwendung
ifxjdbc.jar	enthält Informixtreiber für den Datenbankzugriff der Anwendung
ldapjdk.jar	enthält LDAP Treiber für die Authentifizierung
servlet.jar	enthält Servletklassen zur Anbindung an den servletfähigen Webserver

Eine Implementierung in den Apache Webserver mit Servleterweiterung (Jserv) ist erprobt und sollte somit problemlos möglich sein. Falls etwas fehlschlägt überprüfen Sie bitte die Logdatei des Webserver oder der Servleterweiterung.

Die Einbindung wird durch die Definition von Zonen in Jserv realisiert. Dabei erhält jede Zone eine eigene Konfigurationsdatei (name.properties). Hier eine Beispielkonfiguration.

```
####
# Angabe der Klassen, welche in dieser Zone beheimatet sind
# Hier sind Pfad oder Dateiangaben zulässig,
# wobei Dateiangaben auch Java- oder Ziparchive sein können.
#
repositories=/home/mobiharz/peng/wap.jar
repositories=/home/mobiharz/peng/ifxjdbc.jar
repositories=/home/mobiharz/peng/ldapjdk.jar
repositories=/home/mobiharz/peng/servlet.jar
#
###
# Sollen Klassen oder Dateien zur Laufzeit aktualisiert werden (neu eingelesen) ?
#
autoreload.classes=true
autoreload.file=true
#
###
# Vergabe eines Alias (http://www.host.com/servlet/PeNG
# anstelle von http://www.host.com/servlet/wap.Wrapper)
#
servlet.PeNG.code=wap.Wrapper
#
###
# Festlegung der Konfigurationsdatei von PeNG
#
servlet.wap.Wrapper.initArgs=conf=/home/mobiharz/peng/peng.conf
#
###
# Automatischer Start des Servlets
#
servlets.startup=wap.Wrapper
```

Wie Sie sehen wird der Anwendung eine Konfigurationsdatei bei der Initialisierung durch den Webserver angegeben. Dies ist zwingend notwendig. Sie enthält die

funktionsnotwendigen Daten der Anwendung. Eine Beispielkonfiguration könnte so aussehen.

```
## LDAP Serverkonfiguration

#
# default: ldaphost=
#
ldaphost=info.fh-harz.de

#
# default: ldapport=
#
ldapport=389

## Datenbankkonfiguration

#
# Alle Angaben sind notwendig
#
dbmshost=informix.hs-harz.de
dbmsport=1526
ifxserver=info
user=wap
password=passwort
dbname=httpdb

## Pfad der Applikation
#
# Url-Angabe, die zur Erzeugung dynamischer Links notwendig ist
# evtl. optional, bei Problemen bitte angeben
#
# default: applicationPath=
#
#applicationPath=http://193.175.32.57/peng/PeNG
```

Bei Problemen wenden Sie sich bitte einem Auszug Ihrer Logdatei und den Konfigurationsdateien an edgar.soldin@web.de.

6.3 Marktanalyse Mobiharz (Stand: 17.05.00)

6.3.1 Hardware

Serverhardware

- z.Z. keine besondere Hardware benötigt

Clienthardware

Wichtig für die Entscheidung für ein bestimmtes Gerät ist unter anderem, wie gut diese Geräte unter den Endbenutzern verbreitet sind oder wie diese sich zukünftig auf dem Markt durchsetzen werden. Also sind auch Punkte wie Image des Herstellers und Kaufpreis des Gerätes besonders wichtig.

Nach einer umfangreichen Suche in Fachzeitschriften und im Internet entstand eine Tabelle (siehe Anlage), welche alle aktuellen und teilweise auch zukünftig erscheinenden Endgeräte auflistet. Die Informationen zu den Geräten sind nicht in allen Belangen vollständig, weil die Informationen von den Herstellern zumeist sehr dürftig ausfallen.

So fiel die Entscheidung auf folgende 2 Handies: zum einen auf ein Siemens S35i und zum anderen auf ein Motorola P7389e. Beide Geräte besitzen eine IrDa-Schnittstelle, um drahtlos per Infrarot Daten mit einem anderen Gerät (z.B. einem PDA) zu kommunizieren. Außerdem ist der Kaufpreis für beide Geräte (mit Vertrag) entsprechend gering, was eine hohe Verbreitung unter Mobilfunkkunden garantiert.

Ein Siemens S35i kostet bei D2 Mannesmann zur Zeit mit 2-Jahresvertrag 299,- DM, ein Motorola P7389e bei T-D1 mit 2-Jahresvertrag 99,- DM. Da ein Großteil aller Kunden einen solchen 2-Jahres-Vertrag wählen, sind die eben genannten Preise für uns interessant.

Weiterhin wird auch noch ein drittes Gerät angeschafft, ein Palm 3c, der über IrDa-Schnittstelle problemlos über eines der beiden Handies Daten empfangen oder senden kann, somit kann man mit diesem Palm also auch WAP-Dienste nutzen. Dazu muß allerdings noch ein WAP-Browser installiert werden, welcher durch beiliegende Software gleich mitgeliefert wird. Durch das relativ große Farb-Display können auch spätere WAP-Standards noch problemlos dargestellt werden.

6.3.2 Software

Nokia WAP Server 1.1

Lieferbar ab 2. Quartal 2000

- Connection per SMS und CSD (GSM)
- Über HTTP als Gateway zu Webservern nutzbar
- Stand-alone Apps oder Back-End-System-Connector via Java Servlets (kein Extra-Application Server nötig)
- Static WML oder anderer Content kann lokal abgelegt werden

WAP Funktionalität:

- En-/De-coder für WML / WML-Script bytecode

- Sichere verbindungs-lose / -orientierte Sessions

Server Funktionalität:

- Userauthentifizierung und Zugriffskontrolle
- Applikations- und Content- Hosting
- API zur Back-End-Integration
- Service Provisioning
 - Kann Nokia 6100 und 5100 series mobile phones als SMS bearer einsetzen
(siehe Grafik : WAP Setting Self-Provisioning)
- Fehlermanagement

Content Zugriff:

- WML / WML-Script als Text oder Binärcode
- HTML / Text zu WML
- Zugriff auf HTTP 1.0 / 1.1 Server
- Zugriff auf Local File System
- Zugriff auf Drittanbietersysteme per API

WAP zu E-Mail -Konnektoren

POP3, IMAP4, SMTP, MAPI

User Database

Fungiert als Cookie Proxy

UP.Link Server Suite

UP.Link Gateway:

Web Message Service (WML to SMS) für Non-Wap-Handys`
Security (WTLS) und Authentifizierung

Hardwareanforderungen :

Sun Enterprise Server mit 170 MHz Ultra Sparc Prozessor
256 MB RAM
4 Gig Plattenplatz

Softwareanforderungen :

- Oracle Database Software : 7.3.2.1 7.3.4 8.0.5
Solaris 2.5.1 OS
UP.Link Administration

Ericsson WAP Gateway/Proxy

- bietet Connect vom GSM-Netzwerk zu Internet/Intranet-Applikationsservern
- agiert zum Kabelnetz hin als Client per HTTP / SSL / TCP/IP

Zusätzlich zur Bridgefunktion:

User Authentifizierung
En- / Decodierung von WML-Seiten
WML-Script compilierung
Caching, Service Provisioning, Fehlermanagement

Ericsson WAP Application Server

Plattform für App`s und Content

Runtime Environment Komponenten:

Hardware : Sun Enterprise Server 250

Mind. 50.000 UserRegistrierungen

Erweiterung : mehr Plattenplatz

50 gleichzeitige User

Telefonleitungen und Kapazität des Access Servers erhöhen

benötigt : WAP Gateway und Access Server

User-Registrierung

WAP Mail Client (Mail-Account und App zum Lesen/Schreiben)

HTML to WML

Weitere Services und Content Provider können leicht eingebunden werden

Application Development Environment:

Hardware : Sparc Station 5

API`s für:

UserRegistrierung

UAProf parser (greift Terminaleigenschaften und UserPreferences ab)

HTML to WML

Notification module (für „Push-Technologie“)

HTTP Debugger

WBMP Plugin

6.3.3 SDKs

PHONE.COM UP.SDK

Das SDK beherrscht WML und WML-Script der Version 1.1 und ermöglicht eine Verhaltenssimulation verschiedener Handsets. Bei Neuerscheinungen auf dem Handymarkt erfolgt ein Ausbau des SDK`s einfach über die Erstellung modellspezifischer Plugins. Enthalten sind neben einem HTTP / WAP Client noch Notification Libraries für „Push-Technologie“.

Nokia WAP Toolkit 1.3 beta

Das Toolkit beherrscht WML und WMLScript der Version 1.1 und enthält einen WML und WML-Script Compiler und WYSIWYG-Editor sowie einen WML Browser und Script Interpreter. Weiter enthalten sind Run-time-Libraries, Debugging tools, Java Servlets und ein Wireless Bitmap Editor zur Erstellung von WBMP-Dateien.
- Geräte-Simulatoren

Ericsson WAP IDE

Der WAP-Browser:

simuliert ein WAPgerät und bietet neben realistischen Verhaltensmodellen WML und WML Script Bytecode Interpreter. Der Filezugriff erfolgt über WAP-, HTTP- oder FILE- Clients.

Der Device Designer:

ermöglicht die Erstellung von visuellen Layouts und Verhaltenscharakteristiken zur Simulation verschiedener Endgeräte.

Der Application Designer:
bietet WYSIWYG und Codeerstellung auf zwei Weisen : per Codeeingabe oder Layout (Dual mode Applications source creation). Der kontext-sensitive Editor beherrscht WMLScript und highlighted Code entsprechend der Syntax. Der WML-Browser, mit dem der App Designer arbeitet, ist frei wählbar.

Das Server Toolset:
enthält Perl CGI Libraries zur Erstellung dynamischer WML-Inhalte, der Generierung korrekter HTTP-Header und der Extrahierung von URL-Parametern und -argumenten. Weiter enthalten sind ein Konverter HDML to WML, ein WML Syntax Analyzer und JAFFA Java Servlet Libraries für JavaServer-Umgebungen.
Das Toolset gestattet das Precompiling von WML und WML-Script.

Browser

Weiterhin existieren noch PC-WAP-Browser, wie: WinWap und Wapman, und Handyemulatoren für Ericson R83.

6.3.4 Angebote/Dienste

Inhalte

- Portals (Netzbetreiber, grosse Onlineanbieter)
- Infodienste (Nachrichten, Finanzen, Wetter, Verkehr, Sport)
- Suchmaschinen für WAP-Inhalte
- Service (WAP-Space, WAP-Mail, Terminplaner etc.)
- Unterhaltung (Spiele, Fernsehprogramm, Veranstaltungen)

Navigation

- Hierarchische Navigationsstrukturen mit unterschiedlicher Breite und Tiefe
- zu viele und/oder zu breite Ebenen erschweren die Nutzung der Angebote

Interaktion

Z.Z. nur über Formulareingaben möglich
kaum vorhanden bis auf einige Angebote

Personalisierung

Bei einigen Serviceangeboten und Infodiensten werden nach Login über Nutzerprofile personalisierte Seiten ausgeliefert. Ebenfalls möglich sind technische "Personalisierung" je nach Clienthardware.

Verfügbarkeit

Die Angebote konnten leider nur mit WAP Emulatoren getestet werden und sind nicht immer gleich gut bzw. überhaupt verfügbar. Vermutlich liegt das nicht nur an den PC-WAP-Browsern, so dass beim "echten" WAP-en mit

ähnlichen Problemen zu rechnen ist.

Links

<http://www.wapforum.org/>

<http://www.wapforum.org/what/technical.htm>

http://www.ericsson.de/highlights/wap_technologie.html

<http://www.ericsson.com/letswap/>

<http://www.nokia.de/systemloesungen/wap/index.html>

<http://www.AnywhereYouGo.com/ayg/ayg/Index.po?>

<http://www.wapjag.de/>

Developer zones

<http://www.ericsson.com/developerszone/>

<http://www.forum.nokia.com/waplogin/>

WAPKURS

http://7110.nokia.de/wapkurs/wapkurs_set.html

http://www.AnywhereYouGo.com/ayg/ayg/wap/Article.po?type=WAP_Tutorial

6.3.5 Fazit

- verschieden SDKs, Webserver, Browser und Emulatoren für Win32 vorhanden
- SDKs nur für statische Seiten (keine XML/DB-Unterstützung)
- SDKs unterstützen Entwicklung und Debugging von WML-Script
- Browser und Emulatoren laufen alle noch instabil und liefern keine zuverlässigen Ergebnisse
- bestehende Angebote können bisher nur mit Pc-WAP-Browsern bzw. Emulatoren getestet werden und daher noch nicht ausreichend bewertet werden

6.4 *Klassendiagramm mit Ein-/Ausgabevisualisierung*