



Fachbereich 4: Informatik
Arbeitsgruppe Prof. Hampe
Universitätsstraße 1
56070 Koblenz

Spontaneous Guest Access (SpoGA) & Extended Invitation Management System (E-IMS)

Module
„Veranstaltungsverwaltung“ & „Abrechnung“

Studienarbeit

Betreuer:
Prof. Dr. J. Felix Hampe
Dipl.-Inform. Stefan Stein

eingereicht im Oktober 2007

von

Markus Müller, Raiffeisenstraße 12, 56294 Münstermaifeld
markus.mueller@uni-koblenz.de

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Die Richtlinien der Arbeitsgruppe für Studien- und Diplomarbeiten habe ich gelesen und anerkannt, insbesondere die Regelung des Nutzungsrecht.

Mit der Einstellung dieser Arbeit in die Bibliothek bin ich einverstanden.

ja nein

Der Veröffentlichung dieser Arbeit im Internet stimme ich zu.

ja nein

Koblenz, den _____

Unterschrift

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
Verzeichnis der Quelltexte	VI
1. Einleitung	1
1.1. Ausgangslage	1
1.2. Problematik & Lösungskonzeption	1
1.3. Aufteilung der Arbeiten	2
1.4. Aufbau der Arbeit	3
2. Vorstellung des SpoGA-Systems	4
2.1. Aufbau des Systems	5
3. Vorstellung des E-IMS	7
3.1. Analyse der Anforderungen	7
3.2. Bestandteile des E-IMS	9
3.3. Aufbau des Gesamtsystems	11
3.4. Nutzergruppen und Rollen	12
3.5. Mögliches Geschäftsmodell	14
4. Realisierung des E-IMS	16
4.1. Überlegungen bezüglich der Clients	16
4.2. Überlegungen bezüglich des Servers	16
4.3. Auswahl eines geeigneten Groupware-Systems	18
4.4. eGroupWare	19
4.5. Installation des Basis-Systems	20
5. Entwicklung von Modulen für die eGroupWare	22
5.1. Three-Tier Architektur	22
5.2. Die Verzeichnisstruktur	23
5.3. Die Visualisierungsschicht	24
5.4. Die Datenschicht	28
6. Implementierung	30
6.1. Anforderungen an die Module	30
6.2. Verwendete Entwicklungswerkzeuge	32
6.3. Architektur der Module	32
6.4. Die Veranstaltungsverwaltung	32

6.5. Das Abrechnungsmodul	34
6.6. Struktur der Datenbank	35
7. Anleitung zur Benutzung des E-IMS Systems	37
7.1. Das Modul „Veranstaltungsverwaltung“	37
7.1.1. Die Veranstaltungsübersicht	37
7.1.2. Hinzufügen einer Veranstaltung zur Veranstaltungsübersicht	38
7.1.3. Anlegen einer neuen Veranstaltung	38
7.1.4. Anzeige detaillierter Informationen zu einer Veranstaltung	41
7.1.5. Editieren einer bestehenden Veranstaltung	42
7.1.6. Löschen einer Veranstaltung	43
7.1.7. Buchen von Ressourcen	43
7.1.8. Senden von E-Mails	44
7.1.9. Der Administrationsbereich	44
7.1.10. Optionen des Firmenverantwortlichen	45
7.2. Das Modul „Abrechnung“	46
8. Entfernter Zugriff via Mobiltelefon	48
8.1. Das Szenario	48
8.2. Eingesetzte Techniken und Werkzeuge	48
8.3. Arbeitsweise von XML-RPC	49
8.4. Implementierung einer Beispielanwendung	50
8.5. Anleitung zur Nutzung der Mobiltelefonanwendung	52
9. Abschließende Betrachtungen	54
9.1. Mögliche Erweiterungen	54
9.2. Fazit	54
A. Interview zum geplanten Einsatz eines Groupware-Systems an der Universität Koblenz-Landau	VII
Literaturverzeichnis	IX

Abbildungsverzeichnis

1.	Oberfläche des SpoGA-Systems	4
2.	Architektur des SpoGA-Systems	5
3.	Bestandteile des E-IMS	10
4.	Architektur des SpoGA- & E-IMS Gesamtsystems	12
5.	Rollenverteilung im E-IMS	13
6.	Struktur des eGroupWare-Systems [eGr]	20
7.	Schematische Darstellung der Three-Tier-Architektur	22
8.	Der eTemplate-Editor	26
9.	Einfügen/Modifizieren von Widgets im eTemplate-Editor	26
10.	Use-Case-Diagramme der Module „Veranstaltungsverwaltung“ & „Abrechnung“	30
11.	Vereinfachtes Klassendiagramm des Moduls „Veranstaltungsverwaltung“	33
12.	Klassendiagramm des Moduls „Abrechnung“	34
13.	Datenmodell	35
14.	Hauptmaske des Moduls „Veranstaltungsverwaltung“	37
15.	Hinzufügen neuer Veranstaltungen	38
16.	Anlegen einer neuen Veranstaltung - Schritt 1	39
17.	Anlegen einer neuen Veranstaltung - Schritt 2	40
18.	Beispieltext einer Einladungsmail	40
19.	Anlegen einer neuen Veranstaltung - Schritt 3	41
20.	Anzeigen der Informationen zu einer Veranstaltung	42
21.	Buchen einer Ressource	43
22.	Verfassen einer E-Mail	44
23.	Hinzufügen neuer Firmen	45
24.	Veranstaltungsübersicht im Abrechnungsmodul	46
25.	Rechnungsübersicht	47
26.	Klassendiagramm der Applikation zum entfernten Zugriff via Mobiltelefon	51
27.	(a) Begrüßungsbildschirm (b) Login-Fenster (c) Anzeige von Veranstaltungen	52

Tabellenverzeichnis

1. Berechtigungen der einzelnen Nutzergruppen	14
---	----

Verzeichnis der Quelltexte

1.	HTML-Datei: template.tpl	24
2.	PHP-Datei: executeTemplate.php	24
3.	XML-Datei: test.xet	27
4.	PHP-Datei: executeTemplate.php	28
5.	Definition der Tabelle spoga_events	29
6.	Aussehen einer XML-RPC Anfrage	50

1. Einleitung

Das Einsatzgebiet des im Rahmen von insgesamt drei Studienarbeiten durch Matthias Ehrenstein [Ehr07], Christoph Speich [Spe07] und Markus Müller eingeführten „Spontaneous Guest Access (SpoGA) & Extended Invitation Management System (E-IMS)“ ist im Firmenumfeld angesiedelt. Gästen einer Firma kann durch Nutzung des Systems ohne großen Aufwand ein Zugang zu einem gesicherten, drahtlosen Firmennetzwerk ermöglicht werden, über welches im folgenden der Zugriff auf das Internet erfolgen kann.

Des Weiteren wird die Planung und Durchführung von Veranstaltungen wie insbesondere Konferenzen durch die Nutzung des Systems vereinfacht.

Die folgenden beiden Abschnitte erläutern detailliert die Ausgangslage, die sich hieraus ergebende Problematik sowie die erarbeiteten Lösungen.

1.1. Ausgangslage

Die Nutzung des Internets ist in Zeiten der Globalisierung in der heutigen Geschäftswelt zur Selbstverständlichkeit geworden. Laut einer Studie von Eurostat aus dem Jahre 2006 (vergl. [Eur06]) verfügen in Deutschland etwa 95% aller Firmen über einen Internetzugang. Dieser wird zum Einkauf und Verkauf von Waren, zu Recherche- und natürlich zu Kommunikationszwecken genutzt.

Demzufolge verfügen Unternehmen über eine entsprechende Infrastruktur, die die Arbeitsplätze in Form eines Firmennetzwerks verbindet und hierüber auch die Nutzung des Internets für die Mitarbeiter ermöglicht. Kamen früher vor allem kabelgebundene Netzwerke zum Einsatz, sind in den vergangenen Jahren drahtlose Lösungen erschwinglich geworden. Diese vereinfachen einerseits den Aufbau des Netzes und ermöglichen eine gewisse Mobilität, bieten jedoch andererseits eine breitere Angriffsfläche für potenzielle Eindringlinge. Durch geeignete Sicherungsmaßnahmen versuchen Firmen daher, einen unbefugten Zugriff auf ihr Netzwerk zu verhindern.

Mit der so gewonnenen Sicherheit geht aber oftmals ein Mehraufwand bezüglich der Konfiguration des Netzes sowie der beteiligten Clients einher. Letztere sind vor einer möglichen Nutzung der Netzwerkinfrastruktur in der Regel bei den Netzwerkadministratoren zu registrieren. Dieser Umstand ist insbesondere bei einer lediglich temporären Nutzung des Zugangs, wie beispielsweise durch Firmengäste, nicht akzeptabel.

1.2. Problematik & Lösungskonzeption

Die oben beschriebene Notwendigkeit der Registrierung vor der Netzwerknutzung stellt für die Gäste einer Firma, die nur kurzzeitigen Zugriff auf das Netzwerk benötigen, einen gewissen Komfortverlust dar, da keine direkte Netzwerknutzung möglich ist. Die Administratoren des Firmennetzes werden außerdem doppelt belastet: Neben der Freischaltung des Zugangs ist nach dem Nutzungsende dafür zu sorgen, dass die Zugangsrechte

wieder rückgängig gemacht werden, da diese andernfalls von Eindringlingen missbraucht werden könnten.

An diesem Punkt setzt das „Spontaneous Guest Access“-System, im Folgenden kurz SpoGA genannt, an. Dessen Zielsetzung besteht im Anbieten einer automatisierten Freischaltung der Netzwerkgeräte befugter Gäste durch einen Gastgeber für einen vorgegebenen Zeitraum. Somit kann für bestimmte Geräte komfortabel und ohne großen Aufwand ein temporärer Zugang geschaffen werden, ohne die Sicherheit des Netzes zu beeinflussen. Der Administrator kann vertrauenswürdigen Mitarbeitern das Recht einräumen, Gästen einen Zugang freizuschalten. Durch die Delegation dieser Aufgabe reduziert sich der Arbeitsaufwand des Administrators erheblich und führt somit zu dessen Entlastung. Die Freischaltung geschieht über eine einfache Weboberfläche und erfordert keinerlei technische Kenntnisse mehr.

Mit dem „Extended Invitation Management System“, kurz E-IMS, soll zusätzlich eine Softwarelösung geschaffen werden, die den Verwaltungsaufwand bei der Organisation und Durchführung einer Veranstaltung, wie beispielsweise einer Konferenz, so weit wie möglich reduziert.

Durch die Anbindung an das oben skizzierte SpoGA-System wird darüber hinaus eine automatisierte Freischaltung der Netzwerkzugänge der Konferenzteilnehmer ermöglicht.

Des Weiteren sollen Aspekte wie eine Teilnehmer- und Ressourcenverwaltung berücksichtigt werden. Neben einem automatisierten Versand von Einladungen sollen Teilnehmer außerdem selbständig von ihnen benötigte Ressourcen wie Beamer oder Overhead-Projektoren buchen können. Zu Zwecken der Abrechnung soll zudem eine Übersicht der angefallenen bzw. laufenden Kosten einer Veranstaltung möglich sein. Sämtliche dieser Aufgaben werden bisher in Ermangelung einer entsprechenden Softwarelösung nicht angeboten oder zeitaufwendig manuell durchgeführt.

1.3. Aufteilung der Arbeiten

Die Umsetzung des im vorigen Abschnitt vorgestellten SpoGA-Systems erfolgt im Rahmen der Studienarbeit von Matthias Ehrenstein [Ehr07]. Eine kurze Einführung in die Thematik ist auch im zweiten Kapitel dieser Arbeit zu finden.

Christoph Speich beschäftigt sich in seiner Studienarbeit [Spe07] mit der Umsetzung eines Teilbereichs des E-IMS.

Dieser besteht in der Anbindung des E-IMS an das SpoGA-System sowie der Entwicklung der Ressourcenverwaltung. Ausführlichere Informationen zu den beiden Funktionen finden sich auch im dritten Kapitel dieser Arbeit.

Gegenstand dieser Arbeit ist die Umsetzung des zweiten Teilbereichs des E-IMS, welcher die Realisierung der Veranstaltungsverwaltung sowie der Funktion zum Abrechnen in Anspruch genommener Ressourcen umfasst. Um Nutzern einen mobilen Zugriff auf ihre Veranstaltungsdaten zu ermöglichen erfolgte die prototypische Realisierung einer entsprechenden Anwendung für Mobiltelefone.

1.4. Aufbau der Arbeit

Nachdem dieses erste Kapitel einen kurzen Einblick in das Thema der Arbeit geben konnte erfolgt im weiteren Verlauf eine detaillierte Betrachtung der einzelnen Aspekte.

Dazu präsentiert das zweite Kapitel das SpoGA-System.

Aufbauend hierauf stellt das dritte Kapitel die grundlegenden Konzepte und Funktionen des E-IMS dar.

Die technische Umsetzung dieser Konzeption unter Verwendung eines Groupware-Systems als Ausgangsplattform wird im vierten Kapitel diskutiert.

Allgemeine Grundlagen für die Entwicklung von Erweiterungen für dieses Groupware-System werden anschließend im fünften Kapitel erläutert.

Auf die Implementierung der beiden Module „Veranstaltungsverwaltung“ und „Abrechnung“ als Erweiterungen zu diesem System geht das sechste Kapitel ein.

Eine ausführliche Nutzungsanleitung zu den entwickelten Anwendungen findet sich im siebten Kapitel.

Die Konzeption und Entwicklung der Mobiltelefonanwendung wird im achten Kapitel vorgestellt.

Mit dem abschließenden neunten Kapitel erfolgt eine Betrachtung möglicher Erweiterungen sowie ein Fazit der geleisteten Arbeit.

2. Vorstellung des SpoGA-Systems

Das SpoGA-System bildet den Ausgangspunkt des Gesamtsystems und soll daher in diesem Kapitel kurz beschrieben werden.

Der Nutzen des Systems liegt in der einfachen Freischaltung von Internetzugängen in einem gesicherten, drahtlosen Firmennetzwerk. Benötigt wird diese Funktionalität beispielsweise im so genannten „Berater-Szenario“.

Dieses geht von dem Besuch eines externen Beraters in einer Firma aus, der sich mittels seines Laptops mit dem Internet verbinden möchte. Aufgrund der Sicherung des Netzwerks ist eine Freischaltung des Zugangs durch den jeweiligen Administrator des Firmennetzwerks notwendig.

Durch den Einsatz des SpoGA-Systems kann jedoch ein beliebiger Firmenmitarbeiter, wie etwa der Geschäftspartner des Beraters, in der Rolle des Gastgebers in die Lage versetzt werden, die Freischaltung selbständig zu erwirken.

Ein Vertrauensverhältnis besteht in diesem Fall firmenintern zwischen dem Administrator und dem Gastgeber, also dem Mitarbeiter, der das Recht erhält, Zugänge freizuschalten sowie darüber hinaus zwischen eben diesem Mitarbeiter und seinem Gast, dem externen Berater.

Die Wartung des SpoGA-Systems geschieht in der Regel durch den Administrator des Firmennetzes, der damit die Rolle des SpoGA-Administrators übernimmt.



Abbildung 1: Oberfläche des SpoGA-Systems

2.1. Aufbau des Systems

Das System befindet sich auf einem zentralen Rechner innerhalb einer Firma, dem so genannten SpoGA-Server. Über diesen kann der Gastgeber externen Personen Zugriff auf das Internet gewähren. Zu diesem Zwecke verfügt das SpoGA-System über eine einfach zu bedienende Nutzeroberfläche, die Personen ohne technische Kenntnisse in die Lage versetzt, einen Internetzugang freizuschalten (siehe Abbildung 1). Die Berechtigung hierzu wird vom Administrator des Firmennetzes an vertrauenswürdige Mitarbeiter erteilt.

Die Authentifizierung des Gastes erfolgt mittels eines Freischaltcodes, welcher mit Hilfe einer Kurzmitteilung (SMS) durch das SpoGA-System über ein SMS-Gateway an dessen Mobiltelefon gesendet wird. Die dazu benötigten Mobilfunknummern der Gäste werden vom Gastgeber erfasst.

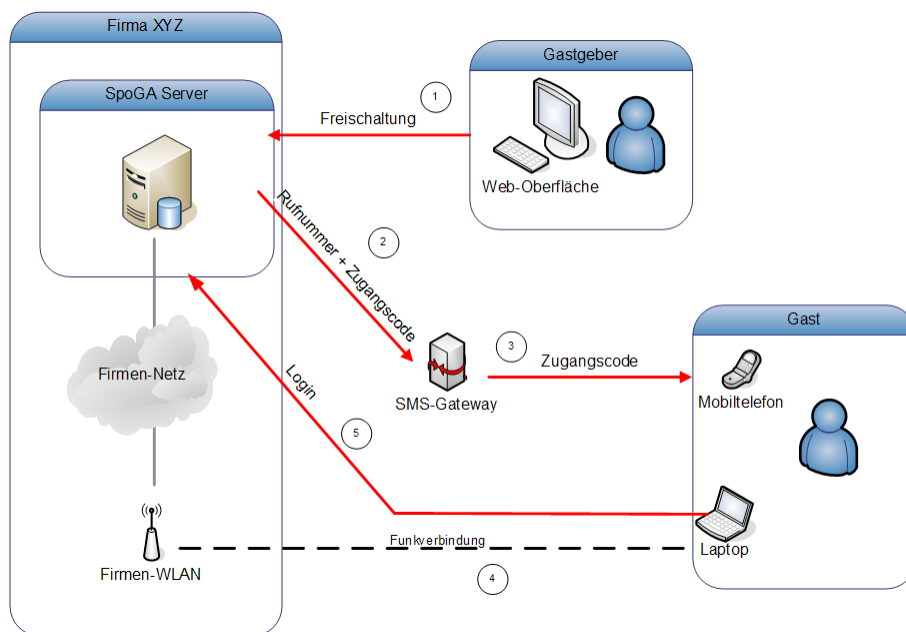


Abbildung 2: Architektur des SpoGA-Systems

Einen Überblick über die Architektur des SpoGA-Systems gibt die oben dargestellte Grafik 2.

Die Freischaltung (1) eines Gastzugangs durch den Gastgeber erfolgt hierbei durch Eingabe der entsprechenden Daten über die Weboberfläche. Diese Daten umfassen die Mobiltelefonnummer des Gastes sowie den für die Nutzung des Zugangs gestatteten Zeitraum. Kurze Zeit vor dem Besuch des Gastes sendet der SpoGA-Server unter Zuhilfenahme eines SMS-Gateways (2) eine SMS mit einem acht stelligen Freischaltcode an die angegebene Mobilfunknummer (3). Der auf dem Firmengelände eingetroffene Gast verbindet sich mit dem Funknetzwerk (4). Beim Versuch, eine Internetseite aufzurufen, wird er automatisch auf eine Startseite des SpoGA-Systems umgelenkt, die die Möglichkeit

bietet, den erhaltenen Freischaltcode einzugeben (5). Nach Überprüfung der Korrektheit dieses Freischaltcodes sowie des entsprechenden Zeitrahmens durch das System kann der Gast nun das Internet in der ihm zur Verfügung stehenden Zeit nutzen. Nach Ablauf der gestatteten Nutzungszeit wird der Zugang automatisch wieder gesperrt und der Gast auf die Startseite des Systems umgeleitet.

Realisiert wird das beschriebene Verhalten durch entsprechende Einträge in der Firewall des SpoGA-Servers, die standardmäßig nicht registrierte Netzwerkgeräte sperrt und solche erst nach Eingabe des korrekten Zugangscodes freischaltet. Genauere Informationen zu diesem Verfahren finden sich in der Ausarbeitung von Matthias Ehrenstein [Ehr07], der für die Umsetzung des Systems verantwortlich zeichnet.

3. Vorstellung des E-IMS

Ein Gastgeber gibt bei der Nutzung des SpoGA-Systems nacheinander die Daten der einzelnen Gäste in Form der Mobilfunknummer sowie der gestatteten Nutzungszeit ein.

Dieses Vorgehen erweist sich insbesondere bei einer großen Anzahl von Gästen als äußerst zeitaufwändig.

Ein solcher Fall wird durch das so genannte „Kongress-Szenario“ beschrieben, welches im Gegensatz zu dem im vorigen Kapitel eingeführten „Berater-Szenario“ von einer auf dem Firmengelände durchzuführenden größeren Veranstaltung, wie eben einem Kongress, ausgeht. Hierbei wünscht nicht mehr nur eine kleine Gruppe von maximal etwa zehn Gästen Zugriff auf das Internet, sondern eine Vielzahl an Personen in Form der Kongressteilnehmer.

Die Nutzung des SpoGA-Systems ist hierbei aufgrund der einleitend erwähnten Einzelerfassung der Daten nicht mehr praktikabel.

Ein weiterer Nachteil besteht in der Tatsache, dass die Daten der Gäste nur für die Zeit der Nutzung gespeichert werden, so dass sie bei einem erneuten Besuch eines Teilnehmers abermals einzugeben sind.

Dieser Problematik wird mit dem Einsatz des E-IMS begegnet. Es stellt eine Ergänzung zum SpoGA-System dar, die dessen Nutzbarkeit auch bei einer Großveranstaltung sicherstellt und darüber hinaus weitere Unterstützung bei der Durchführung und Organisation einer solchen Veranstaltung bietet.

Im Gegensatz zum SpoGA-System wird das E-IMS im Allgemeinen von einem externen Dienstleister angeboten. Das System befindet sich in diesem Fall auf einem zentralen Server außerhalb des Firmengeländes und steht somit unterschiedlichen Firmen gleichzeitig zur Verfügung. Jede das System nutzende Firma erhält einen Firmenaccount, welcher von einem so genannten Firmenverantwortlichen verwaltet wird. Eine genaue Erklärung aller Rollen der beteiligten Akteure findet sich in Abschnitt 3.4.

Erwähnt werden sollte aber in diesem Zusammenhang auch, dass für eine Firma ebenso die Möglichkeit besteht, einen eigenen E-IMS-Server zu betreiben, der exklusiv von ihr genutzt wird. Die Entscheidung hierüber hängt insbesondere von dem zum Einsatz kommenden Tarifierungsmodell des externen Anbieters ab. Rechnet dieser die Nutzung beispielsweise pro Veranstaltung ab, dürfte das Betreiben eines eigenen Servers bei einer großen Anzahl durchgeführter Veranstaltungen sinnvoll sein. Eine Vorstellung möglicher Geschäftsmodelle findet sich in Abschnitt 3.5.

3.1. Analyse der Anforderungen

Der Mehrwert des E-IMS soll in der Vereinfachung der Organisation und Durchführung von Veranstaltungen liegen, woraus Vorteile für beide beteiligten Parteien, Gast und Gastgeber, resultieren.

Eine Ist-Analyse der einzelnen, im Rahmen einer Veranstaltung anfallenden, Prozesse

ergibt ein Optimierungspotenzial, welches in der Automatisierung bisher manuell ausgeführter Aktivitäten liegt.

Nachfolgend findet sich eine Gegenüberstellung des Ist-Zustandes mit dem gewünschten Soll-Zustand.

Ist-Zustand

Ein typischer Ablauf bei der Organisation einer Veranstaltung umfasst etwa die im folgenden genannten Punkte.

Veranstaltungsgäste werden in der Regel vom Gastgeber „manuell“, via Brief, Telefon oder immer häufiger E-Mail, eingeladen.

Für eine Zu- oder Absage der Teilnahme wenden sich die Gäste direkt an den Veranstalter, der diese manuell erfasst.

Gleiches gilt für den Fall, dass ein Teilnehmer bestimmte Ressourcen im Rahmen der Veranstaltung benötigt. Dem Gastgeber obliegt die Aufgabe, das Vorhandensein der gewünschten Ressource zu überprüfen und diese zu reservieren. Kann eine angeforderte Ressource nicht bereit gestellt werden, informiert der Gastgeber den Gast, um diesem eventuell eine Alternative vorzuschlagen. Daraus kann sich mitunter ein aufwändiger und langwieriger Dialog entwickeln.

Durch den Einsatz des SpoGA-Systems ist zwar eine Entlastung des Administrators des Firmennetzes möglich, der Gastgeber zeichnet allerdings nun für die Eingabe der Daten sämtlicher Teilnehmer, im besonderen deren Mobilfunknummer und die gestattete Nutzungszeit, verantwortlich.

Im Falle der kostenpflichtigen Nutzung von Ressourcen ist hierüber zum Veranstaltungsende eine Abrechnung zu erstellen, was eine vorherige Protokollierung der Nutzung voraussetzt

Eine zentrale Übersicht besuchter bzw. angebotener Veranstaltungen sowie angefallener Rechnungen existiert nicht.

Soll-Zustand

Ausgehend von obiger Betrachtung können einige grundlegende funktionelle Anforderungen an das System gestellt werden.

Auf Seiten des Gastgebers gehört zu diesen der Wunsch, Gäste möglichst einfach und mit geringem Aufwand zu einer Veranstaltung einladen zu können. Denkbar ist hier das Auswählen der Teilnehmer aus einem persönlichen Adressbuch mit anschließendem automatischen Versand einer Einladung per E-Mail.

Zu- und Absagen sollen von den Gästen zentral hinterlegt werden können, um dem Gastgeber eine Übersicht des Teilnahmestatus zu ermöglichen.

Die Freischaltung des Internet-Gastzugangs soll vereinfacht werden: Ausgehend von einer Liste der Veranstaltungsteilnehmer soll der Gastgeber für diese einen Internetzugang für einen definierbaren Zeitraum freischalten können. Das Erfassen der zur Zusendung des Freischaltcodes benötigten Mobilfunknummer ist nicht mehr Aufgabe des Gastgebers, sondern erfolgt einmalig durch den Gast. Durch die automatische Übertragung der

entsprechenden Daten an das SpoGA-System entfällt dessen manuelle Einrichtung.

Mit Hilfe einer zentralen Erfassung von Ressourcen können Firmen den Gästen das Reservieren benötigter Ressourcen ohne Zutun des Gastgebers ermöglichen.

Abrechnungen sollen, basierend auf den gespeicherten Nutzungsdaten, zum Veranstaltungsende automatisch erstellt werden können.

Eine Übersicht besuchter bzw. angebotener Veranstaltungen sowie angefallener Abrechnungen vereinfacht Gastgebern und Gästen die Verwaltung von Veranstaltungen.

Auswertung

Auffallend ist die starke Beanspruchung des Gastgebers im „Ist-Zustand“.

Dieser ist zentrale Anlaufstelle für die Gäste und zeichnet für eine Vielzahl der anfallenden Aufgaben verantwortlich.

Verbessern lässt sich diese Situation durch die Einführung eines Softwaresystems, das die Aufgaben des Gastgebers zu einem großen Teil gemäß den Soll-Vorgaben übernimmt.

Dieses System soll zentral von beiden Parteien nutzbar sein: Der Gastgeber hinterlegt sämtliche im Rahmen der Veranstaltung benötigten Informationen, die dann vom Gast abgerufen werden können. Ebenso kann der Gast mit dem System interagieren, um beispielsweise seine Teilnahme zu- oder abzusagen oder Ressourcen zu buchen. Die direkte Kommunikation zwischen Gast- und Gastgeber wird somit deutlich verringert.

Für den Gastgeber ergibt sich als Vorteil eine Entlastung bei der Organisation und Durchführung einer Veranstaltung, da viele Aufgaben automatisiert durchgeführt werden können.

Ein Gast kann mit einem einmalig erstellten Account an unterschiedlichen Veranstaltungen teilnehmen.

Mit dem E-IMS wurde ein Softwaresystem, das die oben beschriebenen Funktionen umsetzt, realisiert. Dessen Konzeption wird in den folgenden Abschnitten erläutert.

3.2. Bestandteile des E-IMS

Die im vorigen Abschnitt analysierten Funktionalitäten lassen sich in die in der nachfolgenden Grafik dargestellten Module untergliedern. Wie aus dieser Grafik ersichtlich ist wurden die beiden Module „SpoGA-Manager“ und „Ressourcenverwaltung“ im Rahmen der Studienarbeit von Christoph Speich umgesetzt [Spe07] während sich diese Studienarbeit mit der Realisierung der Module „Veranstaltungsverwaltung“ und „Abrechnung“ beschäftigte. Diese einzelnen funktionalen Teilbereiche sollen im folgenden näher vorgestellt werden.

Als Ausgangsbasis des E-IMS wurde dabei das Groupware-System „eGroupWare“ verwendet. Dadurch konnte die Neuentwicklung bereits in der Groupware enthaltener Standardfunktionalitäten wie insbesondere die der Nutzerverwaltung, des Kalenders und des E-Mailsystems entfallen. Die unten dargestellten Funktionalitäten wurden somit als Erweiterungen der Groupware, so genannte Module, realisiert.

Im nachfolgenden Kapitel 4 wird detailliert auf die Motivation für den Einsatz einer Groupware als Basis des E-IMS eingegangen sowie die Auswahl eines geeigneten Systems beschrieben.

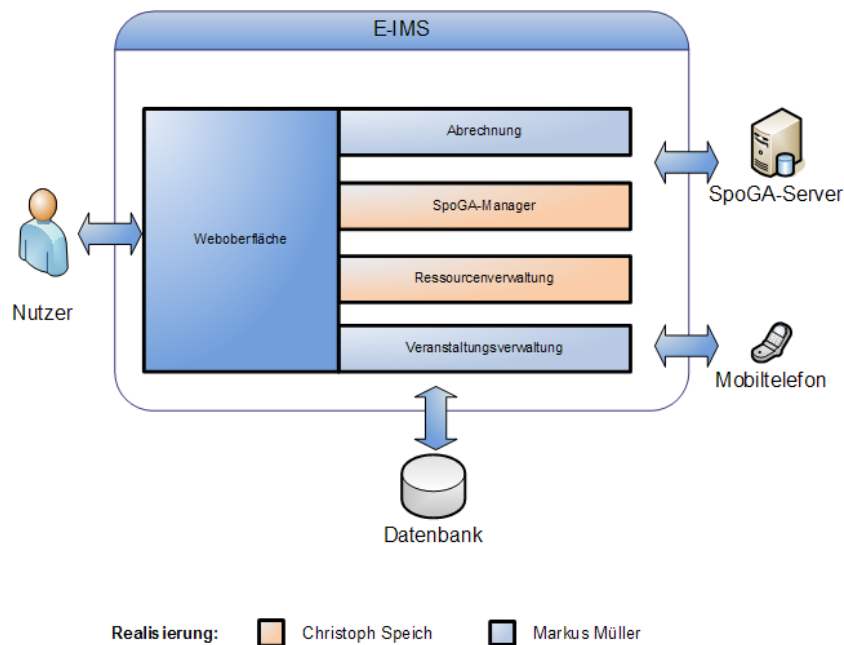


Abbildung 3: Bestandteile des E-IMS

Veranstaltungsverwaltung

Diese erlaubt als zentrales Modul das Planen einer neuen Veranstaltung, was vom Gastgeber der Firma erfolgt. Letzterer legt hierzu die Rahmendaten der Veranstaltung wie deren Titel, den genauen Termin, den Raum und insbesondere auch die Teilnehmer fest. Diese können dabei aus einem von ihm angelegten Adressbuch gewählt werden. Abschließend versendet das System Einladungen an alle Teilnehmer.

Gäste können Details zu den von ihnen besuchten Veranstaltungen einsehen, ihren Teilnahmestatus festlegen sowie benötigte Ressourcen zu einer Veranstaltung buchen.

Neben der Möglichkeit des Zugriffs per Weboberfläche ist dieser auch per Mobiltelefon dank des hierfür entwickelten Clients möglich (siehe Kapitel 8).

Ressourcenverwaltung

Häufig benötigen die Teilnehmer einer Veranstaltung bestimmte Hilfsmittel wie etwa Beamer oder Overheadprojektoren. Die Ressourcenverwaltung stellt für Firmen die Möglichkeit bereit, solche Geräte ihren Veranstaltungsgästen zur Verfügung zu stellen und eventuelle Nutzungsentgelte festzulegen (vgl. [Spe07]). Auf deren Basis kann anschließend eine Abrechnung der Nutzungszeit erfolgen, was durch das im folgenden beschriebene Modul „Abrechnung“ realisiert wird. Das Reservieren der Geräte durch den Gast erfolgt über eine entsprechende Funktion in der Veranstaltungsverwaltung.

Abrechnung

Über diese Funktionalität wird ein Abrechnen der von einem Gast in Anspruch genommenen Ressourcen ermöglicht.

Dazu zählt auch die Abrechnung der Internetnutzung, sofern diese für den Gast kostenpflichtig ist. Entsprechende Angaben sind durch den Gastgeber im Rahmen der Freischaltung des Zugangs zu machen (vgl. [Spe07]). Die Nutzungsdaten werden bei der Rechnungserstellung dynamisch vom SpoGA-Server abgerufen, wozu der Zugriff über einen von diesem bereit gestellten Webservice erfolgt.

Die Nutzungskosten anderer Ressourcen wie Beamer und Overheadprojektoren sind in der Ressourcenverwaltung festzulegen.

Ausgehend von einer Liste der besuchten Veranstaltungen kann ein Teilnehmer die von ihm verursachten Kosten einsehen.

Genauso ist es dem Veranstalter möglich, die im Rahmen einer Veranstaltung angefallenen Gesamtkosten abzurufen.

SpoGA-Manager

Dieser sorgt für die Anbindung des E-IMS an einen firmeninternen SpoGA-Server (siehe Kapitel 2). Hierzu ist ein solcher am E-IMS durch Eingabe verschiedener Parameter wie der IP-Adresse bekannt zu machen. Das Einrichten von Internetfreischaltungen geschieht daraufhin ausgehend von den über die Veranstaltungsverwaltung angelegten Veranstaltungen. Durch das Auswählen einer solchen kann deren Teilnehmern eine bestimmte Nutzungszeit zugeteilt sowie ein Kostenträger für die anfallenden Verbindungsgebühren bestimmt werden. Daneben wird durch den SpoGA-Manager eine Übersicht über die erteilten Freischaltungen angeboten.

Der Gast einer Veranstaltung erhält eine Übersicht der vorhandenen Onlineguthaben sowie die Möglichkeit der Eingabe der Mobilfunknummer, an die der Freischaltcode gesendet werden soll (vgl. [Spe07]).

3.3. Aufbau des Gesamtsystems

In der unten dargestellten Grafik (Abbildung 4) findet sich ein Überblick über die Architektur des Gesamtsystems, welches aus den beiden Systemen SpoGA und E-IMS besteht.

Das im Vergleich zu Abbildung 2 hinzugekommene E-IMS steuert das SpoGA-System bei der Einrichtung. Der Gastgeber interagiert so nicht mehr direkt mit dem SpoGA-System, sondern nutzt das E-IMS als eine Art „Intermediär“.

Über diesen erfolgt die Veranstaltungsabwicklung, zu deren Zweck die Rahmendaten dem System durch den Veranstalter übermittelt werden (1).

Die Gäste werden über ihre Einladung per E-Mail informiert. Nach erfolgter Registrierung besitzen auch sie Zugriff auf das E-IMS (2) und können dort die Daten der Veranstaltung einsehen. Neben der Weboberfläche kann zum Abrufen der Veranstaltungsdaten auch ein Mobiltelefon in Verbindung mit dem prototypisch entwickelten Client (vergl. Kapitel 8) zum Einsatz kommen.

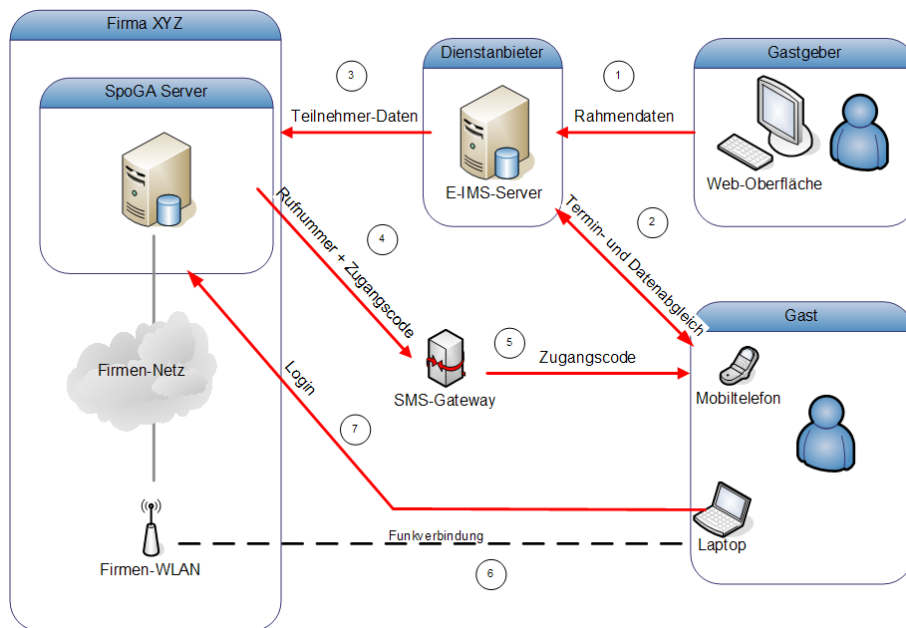


Abbildung 4: Architektur des SpoGA- & E-IMS Gesamtsystems

Daneben haben Nutzer mittels des E-IMS die Möglichkeit, benötigte Ressourcen zu buchen sowie im späteren Verlaufe eine Rechnung anzufordern.

Über das E-IMS erfolgt nun auch die Internetfreischaltung. Ausgehend von einer bestimmten Veranstaltung kann der Gastgeber den Teilnehmern eine Freischaltung für eine bestimmte Zeit und zu eventuell anfallenden Kosten gewähren. Über eine Webservice-Schnittstelle werden die Daten der Teilnehmer wie insbesondere Mobilfunknummer und gestattete Nutzungsdauer automatisiert an den SpoGA-Server übertragen, wodurch ein zeitaufwendiges und fehleranfälliges manuelles Eingeben entfällt (3).

Der weitere Nutzungsablauf entspricht dem bereits aus dem vorigen Kapitel bekannten (vergl. Abschnitt 2.1).

3.4. Nutzergruppen und Rollen

Die Benutzer des E-IMS üben verschiedene Rollen aus und befinden sich daher in unterschiedlichen Nutzergruppen. Diese sind jedoch nicht disjunkt, so dass ein Nutzer durchaus mehreren Nutzergruppen angehören kann.

Durch das Konzept der Nutzergruppen wird einerseits eine Trennung der Befugnisse innerhalb des Systems gewährleistet, so dass beispielsweise nicht jeder Nutzer eine neue Veranstaltung anlegen kann. Andererseits werden Nutzergruppen auch zur Kennzeichnung von Zugehörigkeiten zu bestimmten Nutzerkreisen verwendet, dementsprechend bilden z.B. die Teilnehmer einer Veranstaltung eine eigene Nutzergruppe mit an die Veranstaltung gebundenen Rechten. Ein Nutzer, der an mehreren Veranstaltungen teil-

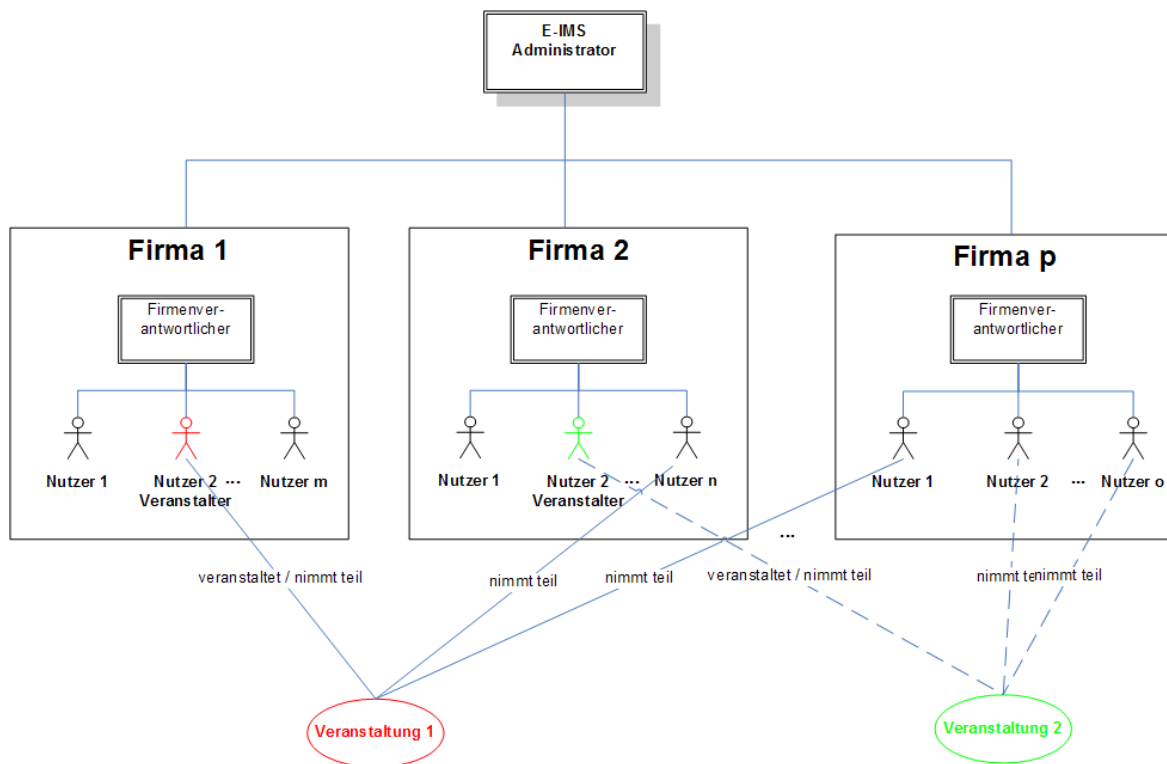


Abbildung 5: Rollenverteilung im E-IMS

nimmt, gehört somit automatisch mehreren Nutzergruppen an.

Eine besondere Rolle im Kreise der Nutzer übernimmt der E-IMS-Administrator, der für die Verwaltung der E-IMS-Installation verantwortlich ist. Er verfügt daher über uneingeschränkten Zugriff auf alle Komponenten des Systems und kann Nutzergruppen sowie darin enthaltene Nutzer einsehen. Für eine Firma, die ein zur Nutzung des E-IMS benötigtes Firmenkonto eröffnen möchte, ist er der direkte Ansprechpartner, da er die zugehörige Firmengruppe anlegt und einen Firmenvantwortlichen (siehe Abbildung 5) festlegt. Dieser ist für die Verwaltung einer Firmengruppe verantwortlich und verfügt über entsprechende Rechte (siehe unten).

Sollte eine Firma einen eigenen E-IMS-Server betreiben, kann der E-IMS-Administrator in einer Doppelfunktion auch die Rolle des SpoGA-Administrators übernehmen. Neben dem E-IMS-Server ist in diesem Fall auch das SpoGA-System zu verwalten.

Standardmäßig befindet sich jeder neue Nutzer nach der Registrierung in der Gruppe „Default“. In dieser kann er an Veranstaltungen, zu denen er vorher eingeladen wurde, teilnehmen, verfügt jedoch ansonsten über keinerlei Rechte.

Zur Verwaltung der einzelnen Firmengruppen existieren die „Firmenvantwortlichen“. Sie können Nutzer zu einer bestehenden Firmengruppe hinzufügen oder aus ihr entfernen sowie insbesondere auch Nutzern den Status eines „Veranstalters“ (siehe unten) zusprechen. Der Status eines „Firmenvantwortlichen“ kann nur vom E-IMS-Administrator

Rollen/Rechte	Nutzer	Firmen- verantwort- licher	Veran- stalter	E-IMS- Admin
Veranstaltungsteilnahme	x			
Veranstaltung anlegen/ mo- difizieren			x	
Firmengruppe verwalten		x		
Veranstalter registrieren		x		
Firmenkonto anlegen				x
Firmenverantwortlichen festlegen				x
eGW-Installation verwalten				x

Tabelle 1: Berechtigungen der einzelnen Nutzergruppen

vergeben werden.

Nutzer, die sich in der Gruppe der „Veranstalter“ befinden, sind berechtigt, neue Veranstaltungen anzulegen und alle damit verbundenen Aktivitäten auszuführen, worunter Aktionen wie das Löschen oder Modifizieren einer selbst angelegten Veranstaltung fallen.

3.5. Mögliches Geschäftsmodell

Das anvisierte Geschäftsmodell geht primär von dem Fall aus, dass das E-IMS von einem externen Dienstleister mehreren Firmen angeboten wird. Es baut auf der in Kapitel 3.4 beschriebenen Rollenverteilung auf. Der Vorteil bei dieser Nutzungsform besteht darin, dass ein Nutzer mit einem einzigen Benutzerkonto an Veranstaltungen verschiedener Firmen teilnehmen kann.

Für die Nutzung des Systems entrichten Firmen und Organisationen eine Nutzungsgebühr an den Dienstleister.

Bei dieser könnte es sich um einen einmaligen Betrag oder aber um monatliche bzw. jährliche Beiträge handeln. Die Höhe dieses Nutzungsentgeltes könnte entweder fix oder variabel gestaltet werden, wobei zur Festlegung eines variablen Beitrages beispielsweise die Anzahl der verwalteten Veranstaltungen herangezogen werden könnte. Auch die Anzahl der Nutzer der jeweiligen Firmengruppe mit dem Nutzerrecht, eine neue Veranstaltung anlegen zu dürfen, könnte als Bemessungsgrundlage dienen.

Da es sich insgesamt um ein modularisiertes System handelt, wäre auch eine Gebühr für die Aktivierung bzw. Nutzung einzelner Module denkbar. So könnte das Modul zum Verwalten von Veranstaltungen als Kernmodul von der oben beschriebenen „Grundgebühr“ abgedeckt werden während hingegen das Abrechnungsmodul oder die Anbindung

an das SpoGA-System als kostenpflichtige Mehrwertdienste durch Zahlung eines separaten Beitrages zu aktivieren wären.

Nicht zu unterschätzen ist für die Nutzer des E-IMS auch die Tatsache, dass es sich um ein externes System handelt und somit weder Hardware- noch Wartungskosten anfallen.

Bei häufiger Nutzung des Systems kann sich das Betreiben eines internen, exklusiv durch die jeweilige Firma nutzbaren, E-IMS-Servers als wirtschaftlich sinnvoll erweisen. In diesem Fall ist die geeignete Serverhardware bereit zu stellen wodurch eventuell Anschaffungskosten anfallen. Darüber hinaus entstehen durch den laufenden Betrieb entsprechende Betriebs- und Wartungskosten. Während ein Nutzer bei einem zentralen System mit seinem Account an Veranstaltungen verschiedener Firmen teilnehmen kann geht dieser Vorteil verloren und eine Teilnahme ist nur an Veranstaltungen der das System betreibenden Firma möglich.

4. Realisierung des E-IMS

Wurden im vorigen Kapitel die funktionalen Aspekte des E-IMS betrachtet sollen im folgenden die technischen Anforderungen sowie die Realisierung dargestellt werden.

4.1. Überlegungen bezüglich der Clients

Beim E-IMS handelt es sich um eine auf einem Server laufende Applikation, auf die mittels Clients zugegriffen wird.

Zwei Alternativen sind bei der Entwicklung dieser Clients denkbar: Zum einen kann ein solcher aus einer für diesen Zweck entwickelten Desktopanwendung bestehen, zum anderen ist eine webbasierte Lösung durch Nutzung eines Internetbrowsers wie dem Microsoft Internet Explorer¹ oder Mozilla Firefox² möglich.

Der Einsatz einer Desktopanwendung bringt einige Vorteile mit sich. Im Laufe der Zeit haben sich für diese Gestaltungskonventionen etabliert, die einem Nutzer die Einarbeitung in eine solche, anfangs unbekannte, Anwendung erleichtern. Auch das Personalisieren einer solchen Anwendung ist im Gegensatz zu einer webbasierten Lösung einfach möglich.

Der Nachteil einer Desktopanwendung liegt allerdings in der mangelnden Flexibilität. Auch wenn dank Technologien wie Java ein Einsatz auf zahlreichen Plattformen ohne besondere Modifikationen möglich ist muss die entsprechende Anwendung auf dem Zielsystem präsent sein. Die Installation und Nutzung könnte beispielsweise aufgrund eingeschränkter Nutzerrechte nicht auf allen einzusetzenden Systemen möglich sein.

Aus diesem Grund fiel die Wahl zugunsten des Einsatzes einer webbasierten Anwendung aus, da der für die Nutzung notwendige Browser heutzutage zur Standardausstattung eines jeden Rechners gehört. Eine Nutzung der Anwendung ist somit von jedem beliebigen PC mit Internetzugang möglich, wodurch ein Nutzer nicht an seinen eigenen Rechner gebunden ist.

4.2. Überlegungen bezüglich des Servers

Die im dritten Kapitel beschriebenen Funktionalitäten, namentlich die Veranstaltungs- und Ressourcenverwaltung, der SpoGA-Manager sowie die Abrechnungsfunktion, sollen Bestandteil der Serverapplikation sein.

Einerseits kann eine solche von Grund auf neu entwickelt werden („from the scratch“), andererseits ist die Nutzung einer existierenden Plattform sowie deren Modifikation und Erweiterung um fehlende Bestandteile denkbar. Letzteres hat den Vorteil, dass eine erneute Entwicklung bereits existierender Funktionalitäten entfallen kann, was eine Zeiterparnis mit sich bringt. Außerdem kann von der Tatsache profitiert werden, dass diese

¹Microsoft Internet Explorer: <http://www.microsoft.com/germany/windows/ie/default.msp>

²Mozilla Firefox: <http://www.mozilla-europe.org/de/products/firefox/>

Bestandteile im Idealfall bereits im Einsatz und damit entsprechend getestet und ausgereift sind.

In Anbetracht dieser Vorteile fiel die Wahl auf die Nutzung einer bereits existierenden Plattform als Grundlage des E-IMS-Servers.

Ausgehend vom gewünschten Funktionsumfang (vergl. Abschnitt 3.2) des E-IMS sind an diese Plattform, unabhängig von deren genauer Ausprägung, folgende minimalen Anforderungen zu stellen:

- Benutzerverwaltung (Personen, Unternehmen)
Diese sollte nach Möglichkeit neben dem Anlegen von Nutzern auch die Erstellung von Nutzergruppen unterstützen, um das in Abschnitt 3.4 diskutierte Rollensystem bestmöglich umsetzen zu können. Vor diesem Hintergrund besteht insbesondere die Forderung, Nutzern die Zugehörigkeit zu mehreren Nutzergruppen zu ermöglichen.
- SMTP-Anbindung zum Versenden von E-Mails
Hierüber sollen beispielsweise die Einladungen zu einer angelegten Veranstaltung versendet werden. Nicht benötigt wird jedoch eine POP- bzw. IMAP-Unterstützung, da das Empfangen von E-Mails nicht zu den vom E-IMS-Provider angebotenen Dienstleistungen gehört und die Nutzer weiterhin über ihre gewohnten E-Mail-Adressen sowie E-Mail-Clients korrespondieren sollen.
- Adressbuch
Dieses ist vor allem für den Gastgeber von Bedeutung, der eine neue Veranstaltung anlegen möchte. Um nicht die Daten jedes einzelnen teilnehmenden Nutzers manuell eingeben zu müssen, sollten diese aus einem bereits vorhandenen Adressbuch ausgewählt werden können. Dieses sollte Importfunktionen für die Übernahme der Daten aus gängigen Systemen wie Microsoft Outlook besitzen oder alternativ seine Daten von einem LDAP-Verzeichnis beziehen können.

Im Bereich der Internetapplikationen finden sich mit Content-Management- (vergl. [EGK06]) und Groupware-Systemen (vergl. [GK07]) zwei Anwendungsformen, die für die Umsetzung des E-IMS-Konzeptes prinzipiell in Frage kommen.

Wie der Name vermuten lässt sind Content-Management-Systeme in erster Linie für das (gemeinschaftliche) Erstellen und Verwalten von Inhalten gedacht. Dementsprechend existiert zwar eine Benutzer- und Rechteverwaltung, Adressbücher für die einzelnen Nutzer sind jedoch in der Regel nicht zu finden.

Groupware-Systeme sind speziell zur Unterstützung kollaborativen Arbeitens gedachte Softwarelösungen, die hierfür eine Vielzahl an Funktionalitäten enthalten. Diese sollen die Kommunikation unter den Nutzern soweit wie möglich vereinfachen und dadurch die Produktivität beim gemeinschaftlichen Arbeiten an Projekten aller Art steigern.

Neben einer Nutzer- und Rechteverwaltung bieten die meisten dieser Systeme Funktionalitäten wie E-Mailversand und -empfang, Diskussionsforen, Dokumentenverwaltungssysteme und gemeinsam nutzbare Kalender. Außerdem existieren häufig verschiedene Schnittstellen, die den Informationsaustausch mit anderen System ermöglichen.

Ebenso wie bei den meisten Content-Management-Systemen ist eine Vergrößerung des Funktionsumfangs durch spezielle Erweiterungen möglich. Ebenso können nicht benötigte Funktionen vollständig oder für einzelne Nutzer bzw. Nutzergruppen deaktiviert werden.

Die Vielzahl an standardmäßig integrierten Funktionalitäten und die damit einhergehende Kompatibilität sowie die Erfüllung der oben beschriebenen Mindestanforderungen führte zu dem Ergebnis, ein Groupware-System als Ausgangsbasis für das E-IMS zu verwenden.

Drei weit verbreitete Lösungen wurden hinsichtlich ihrer Eignung als Grundlage des E-IMS untersucht. Eine kurze Vorstellung dieser Systeme findet sich im nachfolgenden Abschnitt.

4.3. Auswahl eines geeigneten Groupware-Systems

Zur Auswahl eines geeigneten Groupware-Systems wurden drei weit verbreitete Open-Source Lösungen einer genaueren Untersuchung unterzogen. Diese verfügen, wie die meisten dieser Systeme, über die im vorigen Abschnitt gestellten Mindestanforderungen, bestehend aus einer Benutzer- und Rechteverwaltung, der Möglichkeit zum Versenden von E-Mails sowie einem nutzerspezifischen Adressbuch.

Für die genauere Auswahl eines Systems als Plattform des E-IMS wurden daher Aspekte wie Umfang und Güte der erhältlichen Entwicklerdokumentationen, Einfachheit der Installation und Bedienung des Systems sowie nicht zuletzt die zur Entwicklung verwendete Programmiersprache verglichen.

Die drei getesteten Systeme, OpenGroupware, Open-XChange sowie eGroupWare, werden im folgenden kurz vorgestellt.

OpenGroupware [Ope] ist aus einem vormaligen proprietären System der Firma SKYRIX Software AG hervorgegangen. Entwickelt wurde es größtenteils in Objective-C, einer objektorientierten Erweiterung des bekannten ANSI-C.

Open-Xchange [OX] kann als Weiterentwicklung des vormaligen in SUSE Linux enthaltenen Openexchange-Servers angesehen werden. Das System existiert neben der getesteten Open-Source Variante auch in weiteren kommerziellen Versionen, unter anderem in einer Version der Open-Xchange GmbH. Diese bieten neben Dokumentationen und Support erweiterte Funktionalitäten vor allem im Bereich der Konnektoren zu bestehenden Systemen wie Microsoft Exchange³. Das System ist in Java entwickelt.

eGroupWare [eGr] ist eine Entwicklungszweig des Groupware-Systems phpGroupWare⁴. Es zeichnet sich durch eine einfache Installation und eine übersichtliche Weboberfläche aus. Der modulare Aufbau und die Entwicklung in der Skriptsprache PHP ermöglichen ein einfaches Modifizieren der bereits vorhandenen Module sowie ein komfortables Entwickeln eigener Komponenten.

³Microsoft Exchange: <http://www.microsoft.com/germany/exchange/2007/default.mspx>

⁴phpGroupWare: <http://www.phpgroupware.org/>

Nach eingehender Betrachtung dieser Systeme fiel die Entscheidung zugunsten des eGroupWare-Systems aus. Dieses überzeugte durch eine übersichtliche und intuitiv bedienbare Weboberfläche, niedrige Hardwareanforderungen sowie eine einfache Installation. Von der zur Entwicklung verwendeten Programmiersprache PHP wurde zudem eine hohe Effizienz bei der Anwendungsentwicklung bei anfänglich niedrigen Einarbeitungszeiten erwartet.

4.4. eGroupWare

Wie im vorigen Abschnitt beschrieben dient das Groupware-System eGroupWare, im Folgenden kurz eGW genannt, als Ausgangsbasis des E-IMS.

Dabei handelt es sich um einen Groupware-Server, der in seiner Grundkonfiguration bereits zahlreiche Komponenten enthält. Neben den in Kapitel 3 beschriebenen Mindestanforderungen existieren zahlreiche weitere nützliche Module wie ein Kalender sowie eine Ressourcenverwaltung.

Aufgrund der Implementierung in der Skriptsprache PHP läuft das System auf den meisten PHP-fähigen Webservern wie z.B. dem populären Apache HTTP Server⁵, den Microsoft Internet Information Services (IIS)⁶ sowie dem eher unbekanntem Roxen Webserver⁷. Als Speichersystem dient eine Datenbank, wobei zur Zeit sämtliche populären relationalen Datenbanken, namentlich MySQL⁸, PostgreSQL⁹, MaxDB¹⁰, MicrosoftSQL¹¹ sowie Oracle¹², unterstützt werden.

In der unten abgebildeten Grafik wird die Gesamtstruktur des eGW-Systems und insbesondere dessen Schnittstellen beschrieben (siehe auch [eGr]). Die wohl interessanteste Schnittstelle stellt natürlich das native Webinterface dar, mit dem ein direkter Zugriff auf die Groupware möglich ist. Indirekten Zugriff bieten die von den Webservices bekannten Schnittstellen SOAP sowie XML-RPC. Letztere wird im Rahmen dieser Arbeit für den Zugriff mittels Mobiltelefon genutzt (vergl. Kapitel 8).

Die Verwaltung der Nutzerdaten erfolgt wahlweise über ein LDAP-Verzeichnis oder eine der oben aufgeführten Datenbanken. Über diese beiden Systeme kann auch die Authentifizierung erfolgen, wobei als zusätzliche Möglichkeiten hierzu auch der eingesetzte Webserver oder ein IMAP-Verzeichnis in Frage kommen können.

Auch das Betreiben eines Mailserver zum Empfangen bzw. Versenden von E-Mails ist über entsprechende POP3- oder IMAP-Server möglich.

⁵Apache HTTP Server: <http://httpd.apache.org/>

⁶Microsoft Internet Information Services (IIS): <http://www.microsoft.com/windowsserver2003/iis/default.msp>

⁷Roxen Webserver: <http://www.roxen.com/products/webserver/>

⁸MySQL: <http://www.mysql.org/>

⁹PostgreSQL: <http://www.postgresql.org/>

¹⁰MaxDB: <http://www.mysql.com/products/maxdb/>

¹¹MicrosoftSQL: <http://www.microsoft.com/sql/default.msp>

¹²Oracle: <http://www.oracle.com/index.html>

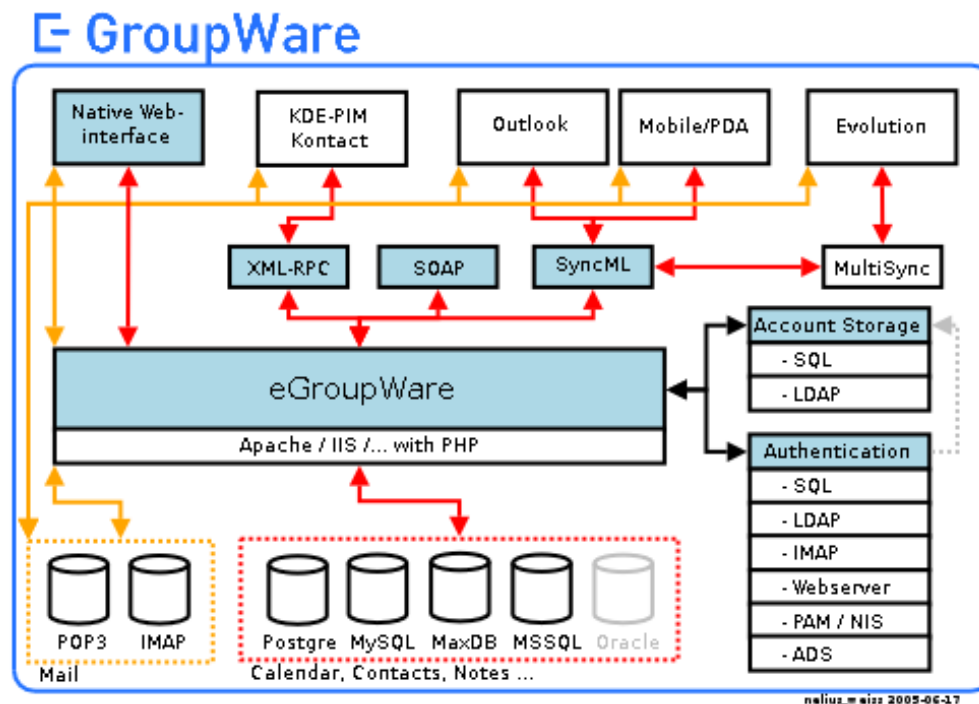


Abbildung 6: Struktur des eGroupWare-Systems [eGr]

4.5. Installation des Basis-Systems

Die erste Installation des eGroupWare-Systems in Version 1.2.104 erfolgte auf einem Rechner mit SUSE Linux 10.0, einem Apache HTTP Server in Version 2.0.54 mit zusätzlicher PHP5-Unterstützung sowie einer MySQL5-Installation in Version 5.0.20a.

Daneben erfolgte zum weiteren Testen und Entwickeln eine lokale Installation auf einem XAMPP-System¹³. Dieses stellt ein Komplettpaket mit sämtlichen zum Betrieb eines Webservers benötigten Komponenten dar und umfasst unter anderem einen Apache-Webserver mit PHP-Unterstützung, eine MySQL-Datenbank sowie einen Mailserver mit SMTP-, POP3- und IMAP-Unterstützung.

Neben diesen beiden zu Entwicklungszwecken gedachten Installationen befindet sich die eGroupWare inklusive der im Rahmen der Studienarbeiten entwickelten Module inzwischen auch auf dem Uniwebpace und ist unter der Adresse <http://www.uni-koblenz.de/~eims> erreichbar.

Die Installation der Groupware gestaltet sich recht einfach. Das heruntergeladene Archiv muss lediglich in den entsprechenden Web-Ordner des Webservers entpackt werden. Bei einem SUSE-System ist dieser standardmäßig unter `/srv/var/www` zu finden, bei einem XAMPP-System im Unterverzeichnis `htdocs`. Die Konfiguration des Systems erfolgt komplett über den Webbrowser. Die benötigte Datenbank wird automatisch ange-

¹³XAMPP: <http://www.apachefriends.org/en/xampp.html>

legt, benötigt werden hier lediglich Nutzernamen und Passwörter eines Datenbanknutzers mit den entsprechenden Rechten. Auch die weitere Konfiguration gestaltet sich relativ einfach und besteht hauptsächlich aus der Angabe der zu verwendenden Verzeichnisse für Datensicherungen, der Anbindung externer Systeme wie LDAP oder E-Mailservern sowie dem Anlegen benötigter Konfigurationsbenutzer.

5. Entwicklung von Modulen für die eGroupWare

5.1. Three-Tier Architektur

Wie bereits in den vorangegangenen Kapiteln beschrieben verfügt das eGroupWare-System über eine große Anzahl an Komponenten für unterschiedliche Aufgaben wie beispielsweise einen Kalender oder ein Adressbuch. Darüber hinaus kann das System jedoch auch um eigene, selbst entwickelte Komponenten erweitert werden.

Die Entwicklung erfolgt dabei in der Skriptsprache PHP gemäß der so genannten „Three-Tier“-Architektur. Dieses in der Softwareentwicklung häufig verwendete Prinzip sieht eine Trennung der Anwendung in Präsentations-, Logik- und Datenschicht vor (siehe beispielsweise [War02]).

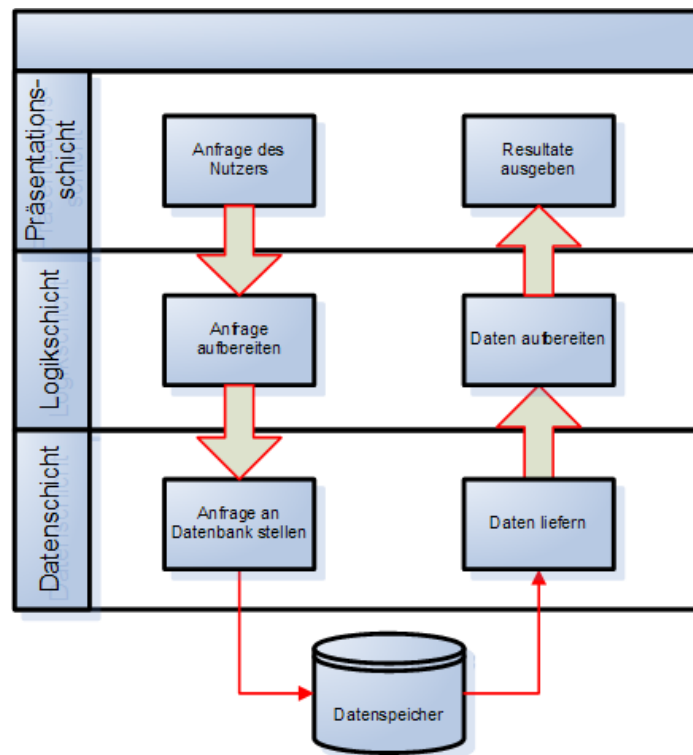


Abbildung 7: Schematische Darstellung der Three-Tier-Architektur

Über die Datenschicht ist die Anwendung mit einem Datenspeicher wie beispielsweise einer Datenbank verbunden. Die daraus extrahierten Daten werden von der darüber liegenden Logikschicht entsprechend der Zielsetzung des Programms weiterverarbeitet und schließlich von der Präsentationsschicht über eine Nutzeroberfläche („Front-End“) an den Nutzer ausgegeben.

Ähnlich wie bei anderen Konzepten der modularisierten Softwareentwicklung bietet

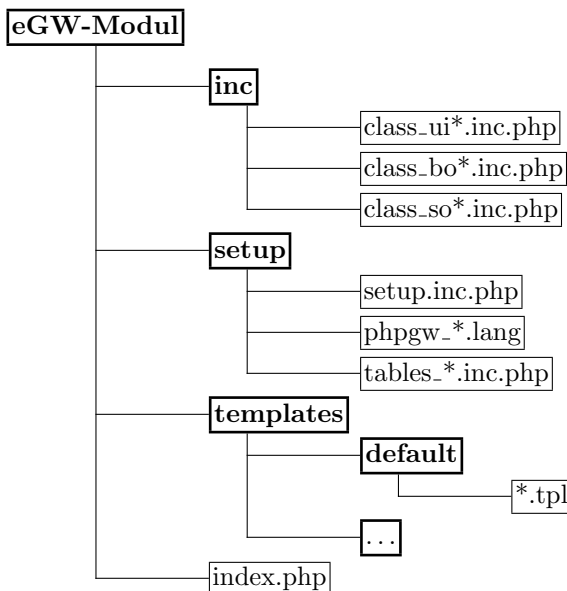
auch diese Form den Vorteil, aufgrund definierter Schnittstellen einzelne Module austauschen zu können ohne dadurch die Funktionsweise der anderen Teile zu beeinträchtigen.

Erfordert beispielsweise der Wechsel der Datenbank eine Neuprogrammierung des entsprechenden Moduls sind hiervon die anderen Module nicht betroffen und können über die definierten Schnittstellen unverändert auch auf das neue entwickelte Modul zugreifen.

Zu beachten ist das der Zugriff auf eine Schicht nur durch die unmittelbar darüber liegende Schicht erfolgen sollte. Dementsprechend wäre also ein Zugriff der Präsentationsschicht auf die Datenschicht nicht gestattet.

5.2. Die Verzeichnisstruktur

Zur Entwicklung und späteren Installation eines eigenen eGW-Moduls wird eine bestimmte Verzeichnisstruktur vorausgesetzt. Diese sollte dem unten dargestellten Muster folgen (vgl. [eGr]).



Hierbei enthält das Unterverzeichnis `/inc` die eigentliche Implementierung, d.h. die Klassen der entwickelten Anwendung. Gemäß dem oben beschriebenen Programmierparadigma sollte es jeweils getrennte Klassen und damit Dateien für jede Schicht geben, wobei eine einzelne Schicht wiederum in mehreren Klassen bzw. Dateien implementiert werden kann.

Die Dateien im Unterverzeichnis `/setup` werden bei der Installation eines Moduls vom eGW-System ausgewertet. Sie enthalten die Tabellendefinitionen der anzulegenden Datenbanktabellen (`tables_*.inc.php`), eventuell vorhandene Übersetzungen in verschiedene Sprachen (`phpgw_*.lang`) sowie Informationen zum Modul selbst wie beispielsweise den Modulnamen, den Autor sowie Informationen über vorhandene Abhängigkeiten mit anderen Modulen (`setup.inc.php`).

Das Verzeichnis `templates` beinhaltet die zur grafischen Darstellung benötigten Templates (siehe Kapitel 5.3).

5.3. Die Visualisierungsschicht

Zur Trennung von HTML- und PHP-Code werden bei der Webentwicklung häufig so genannte Templates verwendet. Ein HTML-Dokument, das an den Browser ausgegeben werden soll, enthält dabei Platzhalter, die vor der eigentlichen Darstellung durch die gewünschten Werte ersetzt werden. Diese Aufgabe übernimmt die Template-Engine, deren Parser das Dokument nach den Platzhaltern durchsucht und die Ersetzungen durchführt. Für den Einsatz mit PHP existieren zahlreiche Alternativen wie beispielsweise „FastTemplates“ [Fas], „smarty“ [sma] sowie „PHPLib“ [PHPb].

Durch die Trennung von Code und Design ergeben sich mehrere Vorteile: Zum einen kann das Aussehen der Webseite durch Verwendung eines anderen Templates einfach verändert werden. Zum Anderen ist durch die saubere Trennung ein unabhängiges Arbeiten an Design und Implementierung möglich, was insbesondere bei großen Projekten mit mehreren Mitarbeitern unterschiedlicher Qualifikationen wie Designer und Entwickler eine Rolle spielen dürfte.

Das nachfolgende Beispiel illustriert die Arbeitsweise von Templates unter Benutzung der PHPLib-Templates (vgl. [Orr]), die auch vom eGroupWare-System verwendet werden.

Listing 1: HTML-Datei: `template.tpl`

```
Mein Name ist {name}.
```

Unter Verwendung des folgenden PHP-Skriptes kann der Platzhalter `{name}` dynamisch durch einen String ersetzt werden.

Listing 2: PHP-Datei: `executeTemplate.php`

```
<?php
include "template.inc";
// späterer Inhalt des Platzhalters
$my_name = "Markus Müller";
// Template-Objekt erzeugen
$tpl = new Template("/home/mydir/mytemplates/");
// erzeugen eines FileHandle für das Template
$tpl->set_file("MyFileHandle","template.tpl");
// Zuweisen des Inhalts von "my_name" an die Template-Variable "name"
$tpl->set_var("name",$my_name);
// parsen des Templates und Speichern des Ergebnisses in "MyOutput"
$tpl->parse("MyOutput","MyFileHandle");
// Ausgabe des geparsten Templates (Inhalt von "MyOutput")
$tpl->p("MyOutput");
?>
```

Neben diesem äußerst einfachen Beispiel können durch die Verwendung von zusätzlichen Elementen wie z.B. geschachtelten Blöcken sehr viel komplexere Template-Strukturen aufgebaut werden, um so etwa komplette Eingabeformulare darzustellen. Der interessierte Leser sei hierbei auf die ausführlichen Dokumentationen in [Orr] und [Sch] verwiesen.

Neben der Verwendung von PHPLib zum Umsetzen von Templates hat sich inzwischen unter den eGroupWare-Entwicklern mit dem eTemplate-System eine andere Variante durchgesetzt. Das gleichnamige Modul für das eGroupWare-System wurde von Ralf Becker, einem der Hauptentwickler, entworfen und umgesetzt (vgl. [eGr]). Es enthält einen grafischen Editor, mit dessen Hilfe die grafische Benutzeroberfläche (GUI) der zu entwickelnden Anwendung erstellt werden kann. Für den Export sowie das Speichern von Templates wird eine stark an XUL (XML User Interface Language) angelehnte Beschreibungssprache verwendet.

Dieser von der Mozilla Foundation entwickelte XML-Dialekt wird zur Darstellung der Oberflächen zahlreicher Programme wie Mozilla Thunderbird und Firefox verwendet (siehe [Moz]). Ziel war auch hier die Trennung von Code und Design sowie eine einfache Portierbarkeit der Oberfläche auf verschiedene Betriebssysteme. Das Aussehen der Oberfläche wird dabei nicht statisch kompiliert sondern von der Rendering-Engine „Gecko“ zur Laufzeit aus den entsprechenden XUL-Dateien ausgelesen und interpretiert, wodurch problemlos eigene Designs entworfen bzw. vorhandene verändert werden können. Das Portieren einer Anwendung auf ein anderes Betriebssystem erfordert lediglich das Anpassen der Rendering-Engine, die eigentlichen Beschreibungsdateien sind hiervon nicht betroffen und müssen somit nur einmal erstellt werden.

Das eTemplate-System organisiert die Anzeige spalten- und zeilenweise. In jede hieraus resultierende Zelle kann in der Regel höchstens ein grafisches Element eingefügt werden. Zur Auswahl stehen dabei unter anderem folgende „Widgets“, wie die grafischen Elemente im eTemplate-Modul genannt werden:

- Textfelder
- Beschriftungen
- Buttons
- Karteikarten (Tabs)
- Ausgabefelder mit speziellen Formatierungen für Uhrzeit und Datum
- Auswahlelemente für Uhrzeit und Datum
- Tabellen
- Nextmatch

Zur Ausgabe von Tabellen ist besonders das letztgenannte Nextmatch-Widget äußerst nützlich, da es im Gegensatz zu „normalen“ HTML-Tabellen einige Zusatzfunktionalitäten mit sich bringt, die andernfalls selbst zu implementieren wären. Hierzu zählt beispielsweise die Möglichkeit, die maximale Anzahl an anzuzeigenden Datensätzen pro Seite festzulegen und zwischen den einzelnen Ergebnisseiten zu blättern.

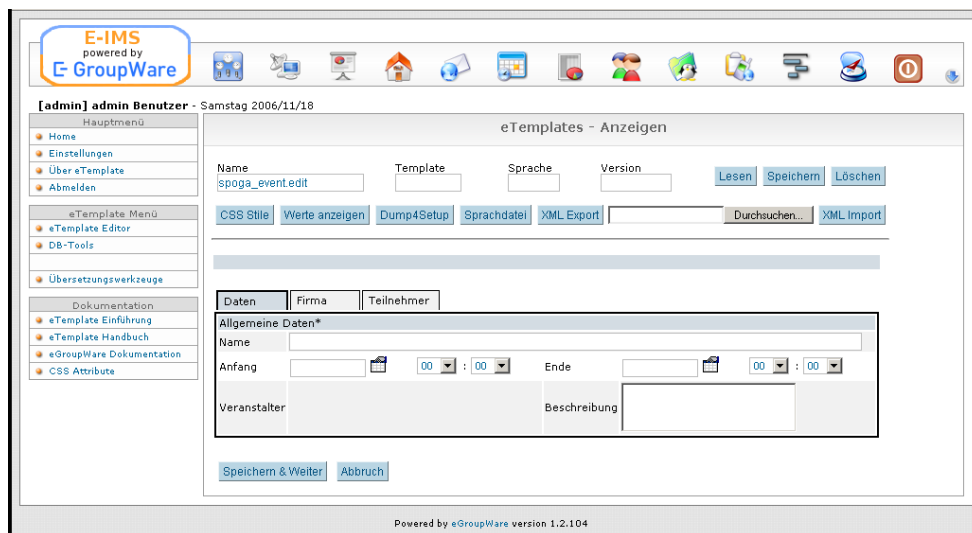


Abbildung 8: Der eTemplate-Editor

Die Entwicklung einer Nutzeroberfläche mit Hilfe des eTemplate-Systems erfolgt in der Regel unter Zuhilfenahme des dafür vorgesehenen grafischen Editors (siehe Abb. 8). Dieser erlaubt das komfortable Konfigurieren der oben vorgestellten „Widgets“ (siehe Abb. 9) sowie das Zusammensetzen mehrerer solcher Elemente zu einer ansprechenden Nutzeroberfläche.

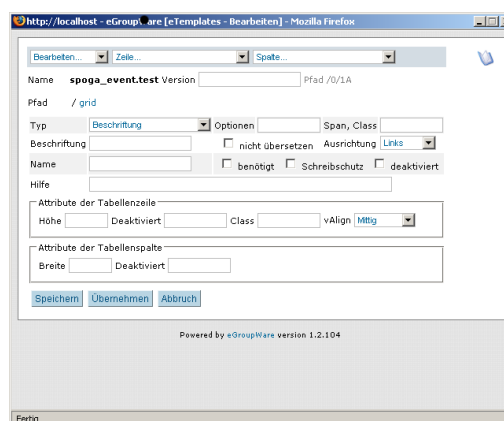


Abbildung 9: Einfügen/Modifizieren von Widgets im eTemplate-Editor

Die Umsetzung des im Zusammenhang mit PHPLib verwendeten Beispiels (siehe oben) lässt sich unter Verwendung des eTemplate-Systems recht einfach durch ein aus einer Zeile und zwei Spalten bestehendes Template realisieren. Dieses enthält zwei Beschriftungs-Widgets, von denen das erste eine fest definierte Beschriftung mit dem Text „Mein Name ist“ besitzt, was durch das Attribut `value` gekennzeichnet wird. Das zweite Beschriftungs-Widget trägt keinen festgelegten Wert, dafür aber einen über das Attribut `id` festgelegten Namen, über dem ihm später Werte zugewiesen werden können.

Die zugehörige XML-Datei ist in Listing 3 dargestellt, wobei die Entwicklung in der Regel vollständig über den grafischen Editor (siehe Abbildungen 8 und 9) erfolgt und keine Eingriffe in der Datei notwendig sind.

Listing 3: XML-Datei: test.xet

```
<?xml version="1.0"?>
<!-- £Id£ -->
<overlay>
  <template id="spoga_event.test" template="" lang="" group="0"
    version="">
    <grid>
      <columns>
        <column/>
        <column/>
      </columns>
      <rows>
        <row>
          <description value="Mein Name ist"/>
          <description id="name"/>
        </row>
        <row>
          <description/>
          <description/>
        </row>
      </rows>
    </grid>
  </template>
</overlay>
```

Die Verarbeitung und Darstellung des Templates geschieht mittels nachfolgendem PHP-Skript:

Listing 4: PHP-Datei: executeTemplate.php

```
<? php
...
$this->etmpl =& CreateObject('etemplate.etemplate',
'spoga_event.test');
...
function test()
{
    $content = array('name' => 'Markus');
    $this->tmpl->exec('spoga_event.test',$content);
}
...
?>
```

Viele der älteren eGW-Module wurden unter Verwendung von PHPLib-Templates entwickelt während inzwischen immer häufiger das eTemplate-System zum Einsatz kommt. Hat man sich erst einmal an die Bedienung des grafischen Editors gewöhnt gelingt das Erstellen auch von komplexen grafischen Oberflächen nicht nur äußerst komfortabel sondern auch relativ zügig und um einiges schneller als dies unter der Benutzung von PHP-Lib möglich wäre.

Nach einem anfänglichen Vergleich der beiden Systeme fiel deshalb die Entscheidung auf das eTemplate-System zur GUI-Gestaltung der zu entwickelnden Groupware-Module.

5.4. Die Datenschicht

Der Zugriff auf eine im Hintergrund befindliche Datenbank geschieht bei eGroupWare mit Hilfe einer Abstraktionsschicht (abstraction layer). Diese ist notwendig, da der Datenbankzugriff in PHP nicht einheitlich ist, es also für die unterschiedlichen, am Markt befindlichen, Datenbanken jeweils eigene Methoden und Funktionen des Zugriffs gibt. Um nicht für jedes dieser Datenbanksysteme (DBMS) eigene Funktionen für den Zugriff vorsehen zu müssen kommt mit ADODB (**A**ctive **D**ata **O**bjects **D**ataBase) eine Abstraktionsschicht zum Einsatz, die ein einheitliches Interface für Datenbankzugriffe bereitstellt und diese dann in die entsprechende Syntax und Semantik des jeweiligen, tatsächlich zum Einsatz kommenden, DBMS umsetzt. Unterstützt werden dabei eine Vielzahl an Datenbanksystemen wie z.B. MySQL, PostgreSQL, MicrosoftSQL und MaxDB.

Die Informationen über die bei der Installation eines Moduls anzulegenden Datenbanktabellen werden in einer standardisierten Syntax abgespeichert, welche unabhängig von einer bestimmten Datenbank ist. Das eGW-System wertet diese Informationen beim Setup des Moduls aus und wandelt sie gemäß den Vorgaben der tatsächlich zum Einsatz kommenden Datenbank um.

Ein Beispiel für dieses Format findet sich in Listing 5 mit der Definition der Tabelle `spoga_events`. Diese enthält sieben Spalten mit den jeweils unter `type` angegebenen Typen. Als Primärschlüssel fungiert, wie mit `pk` für „primary key“ angegeben, die Spalte `event_id`.

Listing 5: Definition der Tabelle `spoga_events`

```
$phpgw_baseline = array(  
  'spoga_events' => array(  
    'fd' => array(  
      'event_id' => array('type' => 'auto', 'precision' => '4',  
        'nullable' => False),  
      'room_id' => array('type' => 'int', 'precision' => '4'),  
      'account_id' => array('type' => 'int', 'precision' => '4'),  
      'description' => array('type' => 'varchar',  
        'precision' => '255'),  
      'name' => array('type' => 'varchar', 'precision' => '255'),  
      'user_group_id' => array('type' => 'int', 'precision' => '4'),  
      'cal_id' => array('type' => 'int', 'precision' => '4')  
    ),  
    'pk' => array('event_id'),  
    'fk' => array(),  
    'ix' => array(),  
    'uc' => array()  
  )  
)
```

Um die Definitionsdateien nicht manuell anlegen zu müssen existiert mit den „db-tools“ aus dem eGW-Modul „eTemplate“ ein Werkzeug zur automatischen Generierung. Damit können unter anderem neue Tabellen angelegt sowie vorhandene bearbeitet werden und zu diesen anschließend die zugehörigen Definitionsdateien angelegt werden.

Zum späteren Zugriff auf die Datenbank stellt das phpGW-API eine öffentliche Variable bereit, über die Anfragen abgesetzt werden können.

6. Implementierung

Ziel dieses Kapitels ist es, einen Überblick über die Funktionsweise der zur Umsetzung der beiden Module „Veranstaltungsverwaltung“ und „Abrechnung“ entwickelten Klassen und Methoden zu geben. Auf eine ausführliche Beschreibung jeder einzelnen Methode wird dabei bewusst verzichtet, da sich entsprechende Kommentare in den Quelldateien finden.

Die ebenfalls im Rahmen dieser Studienarbeit umgesetzte Mobiltelefonanwendung wird in Kapitel 8 vorgestellt.

Im ersten Abschnitt dieses Kapitels werden noch einmal die gewünschten Funktionalitäten anschaulich anhand von Anwendungsfalldiagrammen dargestellt. Nach einer kurzen Vorstellung der zur Entwicklung verwendeten Werkzeuge wird die allgemeine Architektur der beiden Module erläutert bevor die Implementierung in den beiden darauf folgenden Abschnitten 6.4 und 6.5 detaillierter dargestellt wird.

6.1. Anforderungen an die Module

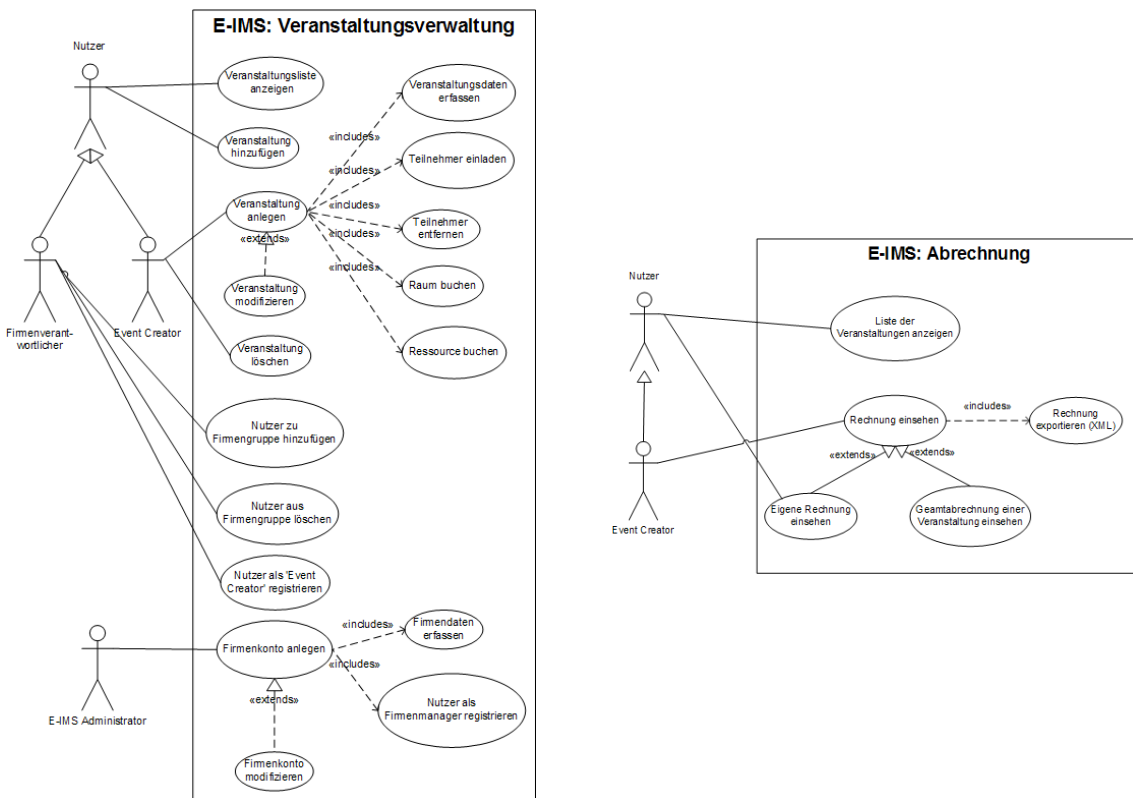


Abbildung 10: Use-Case-Diagramme der Module „Veranstaltungsverwaltung“ & „Abrechnung“

Ausgehend von der im einleitenden Text beschriebenen Konzeption werden von den beiden Modulen „Veranstaltungsverwaltung“ sowie „Abrechnung“ die im Use-Case Diagramm in Abbildung 10 aufgeführten Funktionen erwartet.

Im Rahmen des Moduls „Veranstaltungsverwaltung“ soll der Nutzer die Möglichkeit erhalten, eine neue Veranstaltung anzulegen. Dieser Vorgang umfasst in mehreren Schritten das Eingeben der Veranstaltungsdaten, das Auswählen der einzuladenden Gäste sowie das Selektieren eines Veranstaltungsraumes.

Eine nach diesem Muster angelegte Veranstaltung soll nachträglich editiert werden können, wozu im wesentlichen die obigen Schritte wiederholt werden und dabei die aktuell im System gespeicherten Daten angezeigt werden.

Daneben soll das Entfernen einer im System bereits existierenden Veranstaltung sowie das Buchen von Ressourcen möglich sein.

Ein Gast soll die Möglichkeit erhalten, seine Teilnahme an der betreffenden Veranstaltung zu bestätigen und dadurch diese Veranstaltung zu seiner persönlichen Veranstaltungsübersicht hinzuzufügen.

Diese Veranstaltungsübersicht soll den Ausgangspunkt des Moduls bilden und sämtliche Veranstaltungen tabellarisch mit den dazu vorhandenen Optionen auflisten. Diese Optionen umfassen für den Veranstalter die Möglichkeit, eine Veranstaltung zu löschen oder zu editieren sowie für Gäste die Ressourcenbuchung. Eine ausführliche Vorstellung findet sich in der Benutzeranleitung im nachfolgenden Kapitel.

Zu jeder Veranstaltung soll weiterhin eine Übersichtsseite mit den Veranstaltungsdetails abrufbar sein.

Für den Administrator des E-IMS ist außerdem eine Funktionalität zu implementieren, die es gestattet, Konten für die das System nutzenden Firmen zu eröffnen. Dabei werden firmenspezifischen Daten erfasst sowie der Firmengruppe ein „Firmenverantwortlicher“ (siehe Rollenverteilung 3.4) zugewiesen.

Genau wie das oben beschriebene Modul zur Veranstaltungsverwaltung soll das Abrechnungsmodul eine Veranstaltungsliste als zentralen Ausgangspunkt der Navigation enthalten.

Durch Selektieren einer Veranstaltung kann sich ein Nutzer die in deren Rahmen durch ihn verursachten Kosten anzeigen lassen. Der Gastgeber soll dabei das erweiterte Recht erhalten, die Rechnungen aller Veranstaltungsteilnehmer einzusehen.

Eine Rechnung soll zur etwaigen elektronischen Weiterverarbeitung in einer Datei gespeichert werden können, wozu das CSV-Format (**C**omma **S**eparated **V**alues) [Sha] als am besten geeignet erscheint.

6.2. Verwendete Entwicklungswerkzeuge

Die Entwicklung erfolgte mit dem Open-Source-Framework Eclipse 3.2¹⁴. Durch die zusätzliche Verwendung der Erweiterung „PHPeclipse“¹⁵ wurde dieses um nützliche Funktionen für die Entwicklung mit PHP erweitert, wozu unter anderem die entsprechende Syntax-Hervorhebung zählt. Zur Quellcode-Dokumentation kam die Software „phpDocumentor“¹⁶ zum Einsatz, die ähnlich wie das von der Java-Entwicklung bekannte „Javadoc“¹⁷ ansprechende Quellcode-Dokumentationen erzeugen kann.

6.3. Architektur der Module

Entsprechend der in Abschnitt 5.1 beschriebenen Three-Tier-Architektur bestehen die beiden Module jeweils aus getrennten Klassen für die Präsentations-, Logik- und Datenschicht. Zur Verdeutlichung dieser Unterscheidung tragen die entsprechenden Klassen Präfixe. Dabei steht „so“ für „**S**torage **O**bject“ und kennzeichnet eine zur Datenschicht gehörende Klasse, „bo“ steht für „**B**usiness **O**bject“ und wird somit als Präfix für Objekte der Logikschicht verwendet während Objekten der Darstellungsschicht der Präfix „ui“ für „**U**ser **I**nterface“ vorangestellt wird.

6.4. Die Veranstaltungsverwaltung

Die Gesamtstruktur wird im nachfolgenden UML-Klassendiagramm (Abbildung 11) dargestellt.

Wie hieraus ersichtlich ist wurde die Datenschicht auf mehrere Klassen aufgeteilt. Jede dieser Klassen ist für Zugriffe auf einen bestimmten Teil der Datenbank zuständig.

Die Klasse „so“

Die „Hauptklasse“ für den Datenbankzugriff enthält eine Vielzahl an Methoden für den Zugriff auf unterschiedliche Tabellen.

Hierzu zählen Methoden zum Speichern von Firmendaten, Auslesen verfügbarer Ressourcen wie insbesondere freier Räume, Erzeugen von Kalendereinträgen etc.

Die Klasse „so_participant“

Die Daten der Veranstaltungsteilnehmer befinden sich in der Datenbanktabelle `spoga_participants`. Der Zugriff auf diese Tabelle wird über eine Instanz der Klasse `so_participant` realisiert.

Dementsprechend bietet die Klasse beispielsweise Methoden zum Auslesen der Teilnehmer einer bestimmten Veranstaltung sowie zum Verändern des Teilnahmestatus.

¹⁴Eclipse: <http://www.eclipse.org/>

¹⁵PHPeclipse: <http://www.phpeclipse.de>

¹⁶phpDocumentor: <http://www.phpdoc.de>

¹⁷Javadoc: <http://java.sun.com/j2se/javadoc>

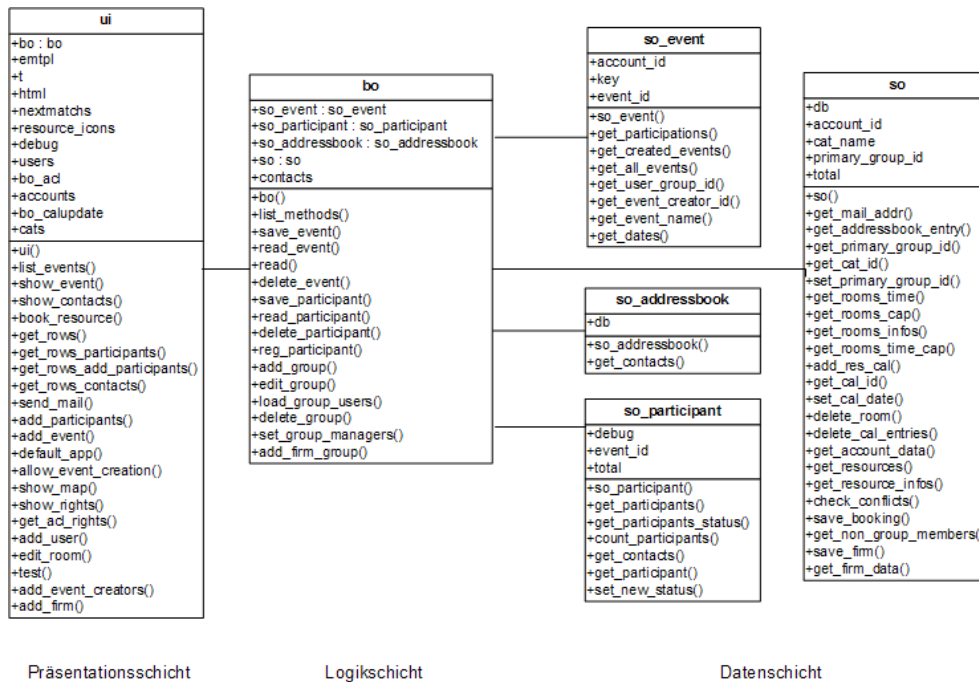


Abbildung 11: Vereinfachtes Klassendiagramm des Moduls „Veranstaltungsverwaltung“

Die Klasse „so_event“

Der Zugriff auf die Tabelle `so_event`, die die Daten der einzelnen Veranstaltungen enthält, erfolgt über eine Instanz der gleichnamigen Klasse `so_event`. Demzufolge bietet sie unter anderem Methoden zum Auslesen der Veranstaltungsteilnahmen eines bestimmten Nutzers sowie der von einem Nutzer initiierten Veranstaltungen.

Die Klasse „bo“

Die Klasse `bo` übernimmt die Rolle der zwischen Daten- und Darstellungsschicht stehenden Logikschicht. Sie stellt insbesondere die Methoden zur Gruppenverwaltung bereit, worunter beispielsweise solche zum Anlegen neuer Gruppen oder Ändern des Gruppenverwalters einer bestimmten Nutzergruppe gehören.

Die Klasse „ui“

Die Rolle der Darstellungsschicht im Kontext der Three-Tier-Architektur wird von der Klasse `ui` übernommen. Viele der dort implementierten Methoden dienen dazu, Daten an die mit dem eTemplate-System (vergleiche 5.3) generierten Formulare und Masken zu senden bzw. von diesen zu erhalten und weiterzuverarbeiten. Hieraus erklärt sich auch das häufige Vorkommen von Methoden mit dem Präfix `get_rows`. Diese liefern die Daten, die das eTemplate-Widget „nextmatch“ (vergleiche Kapitel 5.3) auf dem Bildschirm tabellarisch ausgibt. Daneben existieren beispielsweise Methoden zum Anlegen einer Veranstaltung, Hinzufügen einer Veranstaltung zur Veranstaltungsübersicht, Bu-

chen von Ressourcen usw.

6.5. Das Abrechnungsmodul

Das Abrechnungsmodul ermöglicht die Rechnungsstellung für von Veranstaltungsteilnehmern gebuchte Ressourcen. Dazu gehören insbesondere auch die Kosten, die im Rahmen der SpoGA-Nutzung angefallen sind.

In der Hauptmaske des Moduls werden tabellarisch die Veranstaltungen aufgelistet, an denen der aktuell angemeldete Nutzer teilnimmt bzw. solche, die von ihm erstellt wurden. Über den zur gewünschten Veranstaltung gehörenden Button wird die entsprechende Rechnungsübersicht angezeigt. Analog zum in Kapitel 6.4 beschriebenen Modul „Veranstaltungsverwaltung“ folgt auch dieses Modul dem Konzept der Three-Tier-Architektur und besteht infolgedessen aus drei, die einzelnen Schichten repräsentierenden, Klassen (siehe Abbildung 12). Diese sollen im Folgenden kurz erläutert werden.

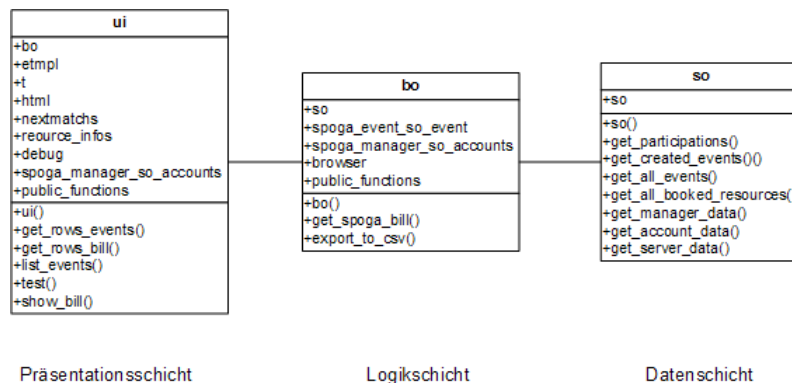


Abbildung 12: Klassendiagramm des Moduls „Abrechnung“

Die Klasse „so“

Die Klasse **so** stellt die Verbindung zu den vom Abrechnungsmodul benötigten Datenbanktabellen her.

Die implementierten Methoden dienen unter anderem zur Abfrage der zu einer Veranstaltung gebuchten Ressourcen sowie zum Auslesen der Konfigurationsdaten eines SpoGA-Servers.

Die Klasse „bo“

Die Kalkulation der Abrechnung erfolgt klassischerweise in der Logikschicht, die durch die Klasse **bo** repräsentiert wird. Die Abrechnungsdaten der SpoGA-Nutzung befinden sich im Gegensatz zu denen der anderen Ressourcen nicht zentral in der Datenbank des E-IMS, sondern verteilt in den Datenbanken der einzelnen SpoGA-Server. Über ein Webservice-Interface ist ein einfacher Zugriff auf diese Daten möglich. Dieser wird

ebenfalls in der Klasse `bo` mit Hilfe eines auf der NuSOAP-Bibliothek aufbauenden Webservice-Clients umgesetzt (siehe auch [Aya]).

Die Klasse „ui“

Die Präsentation der Daten in den zuvor mit dem eTemplate-Editor erstellten Ausgabemasken erfolgt über die Klasse `ui`. Die hierin enthaltenen Methoden sorgen beispielsweise für die Ausgabe der Liste der Veranstaltungen sowie die Anzeige der vom Nutzer gewünschten Rechnung.

Auch hier existieren die von der Klasse `ui` des Moduls „Veranstaltungsverwaltung“ bekannten `get_rows`-Methoden, die für das eingesetzte Nextmatch-Widget benötigt werden.

6.6. Struktur der Datenbank

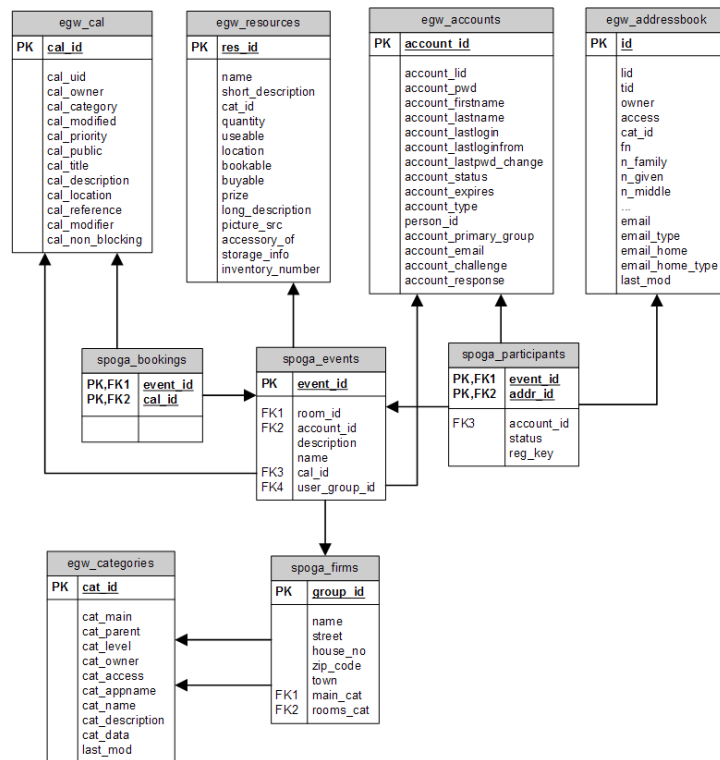


Abbildung 13: Datenmodell

Sämtliche in den beiden entwickelten Anwendungen anfallenden Daten werden in der vom eGroupWare-System bereitgestellten Datenbank gespeichert. Dazu werden einerseits eigene Tabellen von jeder der beiden Anwendungen angelegt sowie andererseits auch die existierenden eGW-Tabellen referenziert, um die dort enthaltenen Daten nutzen zu können. Eine vereinfachte Übersicht des Datenmodells findet sich in Abbildung

13. Bei den Tabellen mit dem Präfix „spoga“ handelt es sich dabei um die neu erstellten Tabellen während solche mit dem Präfix „egw“ die existierenden eGW-Tabellen kennzeichnen.

Die Tabelle `egw_cal` speichert sämtliche innerhalb der Groupware anfallenden Termine, die auch in der Kalenderübersicht eines jeden Nutzers angezeigt werden.

Die Verwaltung der Nutzerkonten erfolgt mittels der Tabelle `egw_accounts`, in der sich die entsprechenden nutzerspezifischen Daten wie etwa Vorname und Name sowie E-Mail-Adresse finden.

Das Adressbuch eines jeden Nutzers wird in `egw_addressbook` gespeichert. Diese Tabelle umfasst dabei typische Daten wie Vorname und Name eines Kontaktes, seine Anschrift sowie die E-Mail-Adresse.

Informationen zu den vorhandenen Ressourcen werden in der Tabelle `egw_resources` gesichert, wobei hierbei, beispielsweise zur Angabe einer Ausleihdauer, von der Kalendertabelle `egw_cal` Gebrauch gemacht wird.

Die von der eGroupWare verwalteten Ressourcen gehören in der Regel einer bestimmten Kategorie an, deren Daten in der Tabelle `egw_cat` gespeichert werden. Um die Ressourcen der am System angemeldeten Firmen möglichst geordnet erfassen zu können werden diverse Standardkategorien wie „Räume“ oder „Beamer“ für jede neu hinzugefügte Firma erstellt.

Die Daten der das E-IMS nutzenden Firmen finden sich in der Tabelle `spoga_firms` wieder. Dazu gehören Name und Anschrift der Firma sowie Referenzen auf die Hauptkategorie sowie die zur Erfassung der bereitgestellten Räume verwendete Kategorie.

Für die Speicherung der Daten einer Veranstaltung zeichnet `spoga_event` verantwortlich. Gespeichert werden hier der Name der Veranstaltung sowie die zugehörige Kurzbeschreibung. Daneben wird der in der Ressourcentabelle `egw_resources` gespeicherte Veranstaltungsraum, die Daten des Organisers, der zur Veranstaltung gehörende Kalendereintrag sowie die Firmengruppe des anlegenden Organisers gesichert.

Die an einer Veranstaltung teilnehmenden Nutzer werden mittels der Tabelle `spoga_participants` gespeichert. Dieser Tabelle obliegt die Aufgabe, die im System eindeutige `account_id` eines jeden Nutzers mit der im Adressbuch eines Nutzers vorhandene `addr_id` zu „verknüpfen“, auf deren Basis eine Einladung erfolgt. Daneben wird die Veranstaltung, für die die Teilnahme gelten soll, referenziert sowie der Teilnahme-status und ein vom Nutzer einzugebender Registrierungsschlüssel gespeichert.

7. Anleitung zur Benutzung des E-IMS Systems

7.1. Das Modul „Veranstaltungsverwaltung“

Das im Folgenden beschriebene Modul ist das eigentliche „Kernmodul“ des E-IMS Systems. Neben der Anzeige sowie dem Anlegen von Veranstaltungen erlaubt es das Registrieren neuer Firmen im System durch den E-IMS-Administrator sowie das Verwalten der Firmengruppen durch die jeweiligen Firmenverantwortlichen.

7.1.1. Die Veranstaltungsübersicht

Die Hauptmaske des Moduls „Veranstaltungsverwaltung“ (siehe Abbildung 14) zeigt in einer Listenansicht die Veranstaltungen des angemeldeten Nutzers an. Dabei kann diese Liste nach drei Kriterien gefiltert werden:

- „Teilnahmen“
zeigt nur solche Veranstaltungen an, an denen der Nutzer teilnimmt.
- „Angelegte“
zeigt nur die Veranstaltungen an, die vom Nutzer selbst angelegt wurden.
- „alle“
Kombination aus den beiden obigen Ansichten, d.h. es werden sowohl die Veranstaltungen angezeigt, an denen der Nutzer teilnimmt als auch solche, die vom Nutzer selbst angelegt wurden.

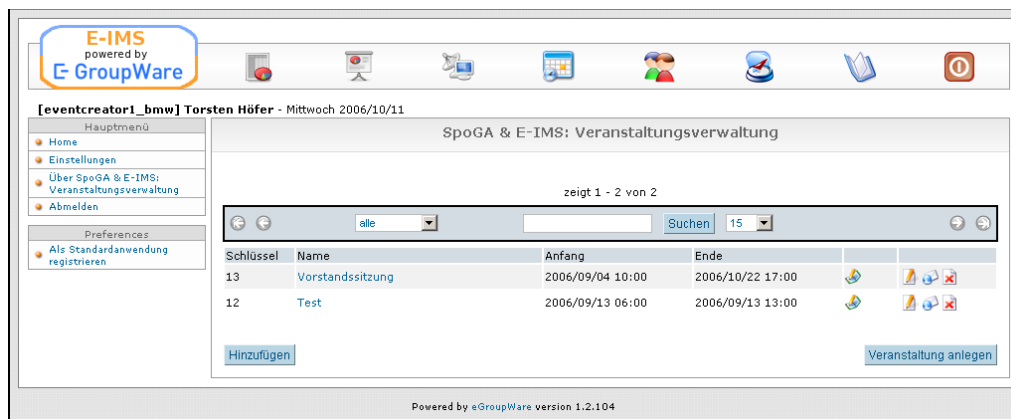






Abbildung 14: Hauptmaske des Moduls „Veranstaltungsverwaltung“

Wie in allen im folgenden beschriebenen Listenansichten kann die Anzahl der pro Seite maximal angezeigten Datensätze durch die Auswahlbox am rechten oberen Rand eingestellt werden. Übertrifft die Anzahl an gefundenen Datensätzen diesen Wert erfolgt

eine seitenweise Darstellung der Ergebnisse, wobei mit den Pfeilbuttons zwischen den Seiten geblättert werden kann. Mit Hilfe der Buttons  bzw.  gelangt man dabei zur vorigen bzw. nächsten Seite der Ergebnisliste sowie mittels  bzw.  zur ersten bzw. letzten Seite der Liste.

Mittels des Suchfeldes kann nach bestimmten Veranstaltungen gesucht werden, wobei die Felder „Name“ und „Beschreibung“ nach dem eingegebenen Suchbegriff durchsucht werden.

7.1.2. Hinzufügen einer Veranstaltung zur Veranstaltungsübersicht

Mit Hilfe des in einer Einladungsmail (siehe Abbildung 18) enthaltenen Schlüssels kann eine Veranstaltung zur Veranstaltungsübersicht eines Nutzers hinzugefügt werden.

Nach Betätigen des Buttons [Hinzufügen](#) in der Veranstaltungsübersicht (siehe Abbildung 14) erscheint dazu das in Abbildung 15 dargestellte Fenster, in dessen einziges Eingabefeld der Veranstaltungsschlüssel einzugeben ist. Durch Betätigen des Buttons [Hinzufügen](#) wird das Fenster geschlossen und die Veranstaltungsübersicht automatisch aktualisiert, so dass nun auch die gewünschte Veranstaltung enthalten ist. Zu beachten ist das der Teilnahmestatus standardmäßig auf „Unbekannt“ gesetzt wird. Ein Ändern ist jederzeit in der detaillierten Ansicht der Veranstaltung unter der Registerkarte „Status“ möglich (siehe auch 7.1.4).

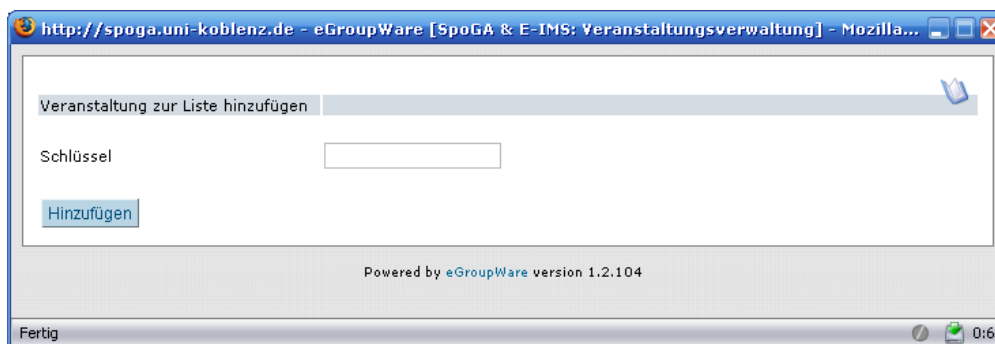


Abbildung 15: Hinzufügen neuer Veranstaltungen

7.1.3. Anlegen einer neuen Veranstaltung

Das Anlegen einer neuen Veranstaltung kann nur von Nutzern durchgeführt werden, die über die entsprechenden Rechte verfügen, also vom Firmenverantwortlichen als „Veranstalter“ registriert wurden. Für diese Nutzer existiert in der Veranstaltungsverwaltung der Button [Veranstaltung anlegen](#).

Nach dessen Betätigung öffnet sich ein neues Browserfenster, das dem Nutzer in drei Schritten das Anlegen einer Veranstaltung ermöglicht. Im ersten Schritt (Abbildung 16) sind hierbei folgende Daten einzugeben:

Abbildung 16: Anlegen einer neuen Veranstaltung - Schritt 1

- Name:
Kennzeichnet den Namen der Veranstaltung, z.B. „Jahresversammlung“. Dabei sollte ein möglichst aussagekräftiger Titel gewählt werden, da dieser später auch in der Veranstaltungsübersicht erscheint.
- Anfang:
Anfangsdatum und -zeit der Veranstaltung.
- Ende:
Enddatum und -zeit der Veranstaltung.
Zu beachten ist hierbei, dass der Anfangszeitpunkt vor dem Endzeitpunkt liegen muss.
- Veranstalter:
Hier werden vom System automatisch Vorname, Name und E-Mail-Adresse des Veranstalters eingetragen, die dann später für die Teilnehmer sichtbar sind.
- Beschreibung:
In diesem Textfeld kann eine kurze Beschreibung der Veranstaltung erfolgen, die ergänzend zum Titel der Charakterisierung dienen soll.

Die Daten der zweiten Registerkarte „Firma“ werden vom System automatisch eingetragen. Hier erscheinen Informationen zu der Firma, der der Veranstalter angehört. Diese Daten wurden bei der Registrierung der Firma vom E-IMS-Administrator eingegeben und können nicht von einem Benutzer verändert werden.

Durch Betätigen des Buttons **Speichern & Weiter** gelangt man zur Auswahl der Veranstaltungsteilnehmer. Hierbei erscheinen alle Kontakte aus dem Adressbuch des Veranstalters. Teilnehmer sind mit einem Haken in der entsprechenden Checkbox zu markieren (siehe Abbildung 17).

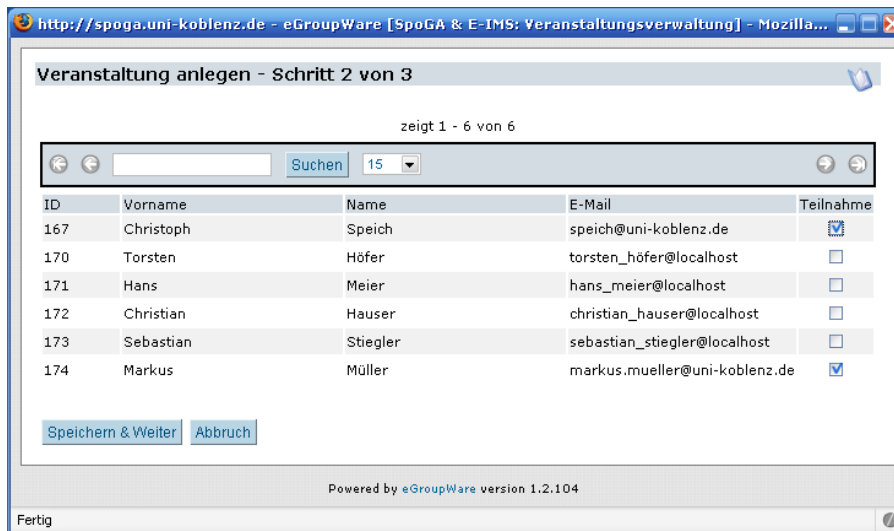


Abbildung 17: Anlegen einer neuen Veranstaltung - Schritt 2

Nachdem die Auswahl der Veranstaltungsteilnehmer mit **Speichern & Weiter** gespeichert wurde versendet das System im Hintergrund die entsprechenden Einladungsmails, deren Text in Abbildung 18 exemplarisch dargestellt ist.

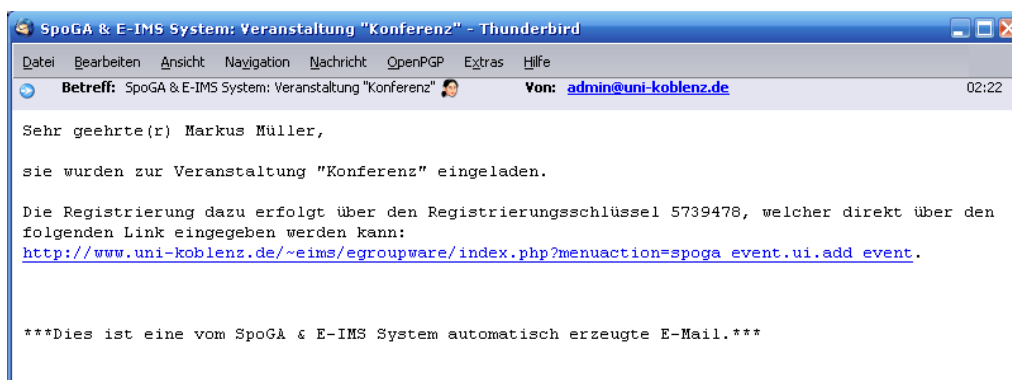


Abbildung 18: Beispieltext einer Einladungsmail

Schlägt das Versenden einer bzw. mehrerer Mails fehl, informiert das System hierüber in der nachfolgenden Maske, die den Abschluss der Veranstaltungserstellung darstellt.

In dieser erfolgt die Auswahl eines geeigneten Raumes (siehe Abbildung 19), wobei nur solche Räume aus der Ressourcenverwaltung angezeigt werden, die zum entspre-



Abbildung 19: Anlegen einer neuen Veranstaltung - Schritt 3

chenden Termin nicht belegt sind und deren Kapazität für die Anzahl der ausgewählten Teilnehmer ausreichend ist. Mit „Übernehmen“ wird die Auswahl eines Raumes im System gespeichert. Durch Betätigen des Buttons „Schließen“ gelangt man zur Veranstaltungsübersicht zurück, die nach einer automatischen Aktualisierung die neu angelegte Veranstaltung anzeigt.

7.1.4. Anzeige detaillierter Informationen zu einer Veranstaltung

Zu jeder in der Veranstaltungsübersicht vorhandenen Veranstaltung können detaillierte Information durch Anklicken des als Link markierten Veranstaltungsnamens angezeigt werden. Ein Beispiel findet sich in Abbildung 20.

Über die fünf dargestellten Registerkarten „Daten“, „Firma“, „Raum“, „Teilnehmer“ sowie „Status“ gelangt man zu den zur entsprechenden Veranstaltung vorhandenen Informationen.

Hervorzuheben ist hier insbesondere der Punkt „Status“, der dem Teilnehmer das Ändern seines Teilnahmestatus an der Veranstaltung gestattet. Dieser wird standardmäßig auf den Wert „Unbekannt“ eingestellt, zusätzlich stehen „Bestätigt“ zum Bestätigen der Teilnahme sowie „Abgesagt“ zum Absagen einer Veranstaltungsteilnahme zur Verfügung. Mit diesen Daten wird dem Veranstalter ermöglicht, die Anzahl der tatsächlich teilnehmenden Personen zu kalkulieren.

Unter dem Punkt „Teilnehmer“ stehen diese drei Status des Weiteren als Filteroptionen für die Liste der Veranstaltungsteilnehmer zur Verfügung.

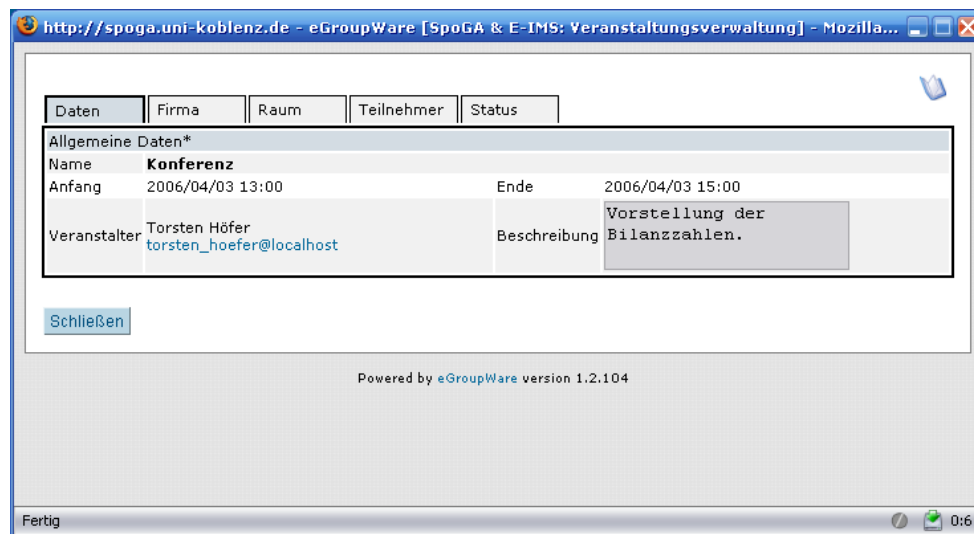



Abbildung 20: Anzeigen der Informationen zu einer Veranstaltung


7.1.5. Editieren einer bestehenden Veranstaltung

Die Daten einer bereits angelegten Veranstaltung können nachträglich vom Veranstalter geändert werden. Hierzu steht der Button „Veranstaltung editieren“  zur Verfügung. Das Editieren der Daten erfolgt dabei analog zum Anlegen einer neuen Veranstaltung in drei Schritten bzw. drei Eingabemasken:


- Editieren der allgemeinen Veranstaltungsdaten
Zusätzlich zu den beiden vom Anlegen einer Veranstaltung bekannten Registerkarten „Daten“ und „Firma“ steht nun eine dritte Registerkarte „Teilnehmer“ zur Verfügung. Über diese können Veranstaltungsteilnehmer von der entsprechenden Veranstaltung „ausgeladen“ werden, d.h. sie werden aus der Liste der Veranstaltungsteilnehmer gelöscht. Damit einher geht das Entziehen der entsprechenden Rechte als Veranstaltungsteilnehmer wie z.B. das Buchen von Ressourcen. Des Weiteren erscheint die Veranstaltung nicht länger in der Liste der besuchten Veranstaltungen des „ausgeladenen“ Nutzers.
- Hinzufügen neuer Teilnehmer
Hierüber können weitere Teilnehmer zur aktuellen Veranstaltung hinzugefügt werden. Dabei erscheinen in der Auswahlliste alle Personen aus dem Adressbuch des Veranstalters, die nicht bereits an der Veranstaltung teilnehmen. Diese Aktion ist zeitlich nicht beschränkt, Teilnehmer können somit auch zu einer bereits laufenden Veranstaltung eingeladen werden.
- Ändern der Raumbuchung
Der aktuell für die Veranstaltung ausgewählte Raum wird am Seitenanfang unter

„Aktuelle Auswahl“ angezeigt. Soll dieser geändert werden geschieht dies durch das Selektieren eines Raumes aus der angezeigten Auswahlliste und anschließendes Betätigen des Buttons **Übernehmen**.

7.1.6. Löschen einer Veranstaltung

Das Löschen einer existierenden Veranstaltung geschieht durch Betätigen des Buttons  in der Veranstaltungsübersicht und kann nur vom Veranstalter selber durchgeführt werden. Eine beim Anlegen der Veranstaltung erfolgte Raumbuchung wird hierbei ebenfalls gelöscht.

7.1.7. Buchen von Ressourcen

Von jedem Teilnehmer einer Veranstaltung können zu dieser entsprechende Ressourcen gebucht werden. Dies geschieht über den in der Veranstaltungsübersicht dargestellten Button  (vergl. Abbildung 14).

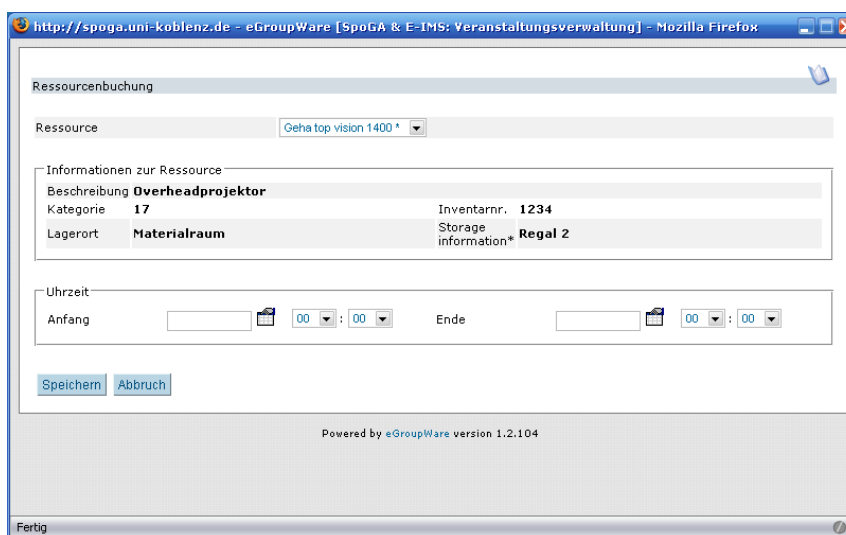





Abbildung 21: Buchen einer Ressource

Nach dessen Betätigung erscheint das in Abbildung 21 dargestellte Browserfenster. Über die Auswahlbox kann hier die gewünschte Ressource ausgewählt werden, zu der eine eventuell vorhandene Beschreibung sowie, falls erfasst, weitere Informationen wie der aktuelle Lagerort angezeigt werden. Nach Angabe des Ausleihzeitraumes, der innerhalb des Veranstaltungszeitraumes liegen muss, kann die Buchung durch Betätigen des Buttons **Speichern** gespeichert werden.

Die auf diese Weise buchbaren Ressourcen müssen vom Firmenverantwortlichen zuvor über die Ressourcenverwaltung  des Groupware-Systems eingetragen worden sein (siehe [Spe07]).

7.1.8. Senden von E-Mails

Über den in der Veranstaltungsübersicht angezeigten Button  wird dem Veranstalter das Verfassen und Versenden einer E-Mail an sämtliche Veranstaltungsteilnehmer ermöglicht (siehe Abbildung 22).

In der Zeile „Betreff“ ist der Betreff der E-Mail anzugeben. Das Verfassen des Textes wird durch einen komfortablen Editor unterstützt, der auch das Formatieren des Textes im HTML-Format erlaubt. Nach dem Betätigen des Buttons  wird die E-Mail automatisch an alle Veranstaltungsteilnehmer versendet. Beim Versenden aufgetretene Fehler, wie beispielsweise der Versand an eine ungültige E-Mail-Adresse, werden durch eine Fehlermeldung im gleichen Fenster kenntlich gemacht.

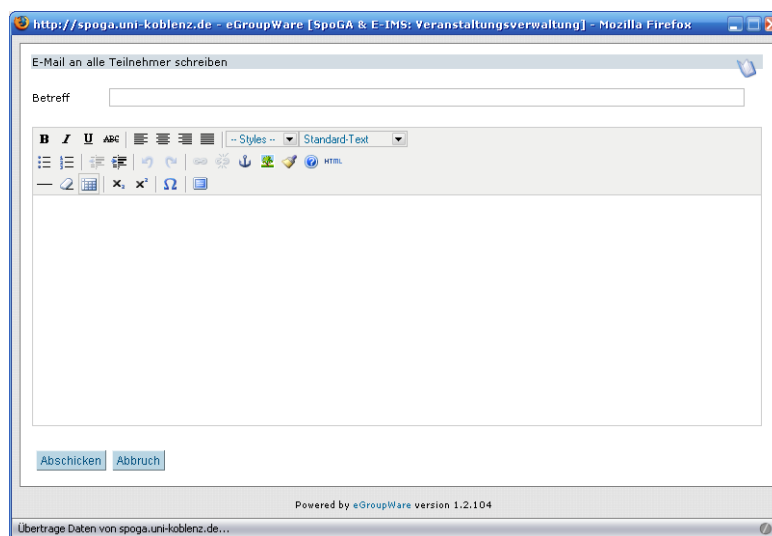



Abbildung 22: Verfassen einer E-Mail

7.1.9. Der Administrationsbereich

Im Administrationsbereich besteht für den E-IMS-Administrator die Möglichkeit, neue Firmen zu registrieren. Dies geschieht über die Veranstaltungsverwaltung mittels des in Abbildung 23 dargestellten Menüpunktes.

In der ersten Registerkarte „Daten“ sind hierbei die Firmendaten einzugeben, in der zweiten Registerkarte „Manager“ wird der zuständige Firmenverantwortliche aus der Liste der im System registrierten Benutzer ausgewählt. Durch Betätigen des Buttons  wird die Firma im System registriert.

Die hier angegebenen Firmendaten werden beim Anlegen einer Veranstaltung automatisch übernommen, wie in Abschnitt 7.1.3 beschrieben, und sind damit für die Veranstaltungsteilnehmer über die entsprechende Übersichtsseite der Veranstaltung abrufbar (siehe 7.1.4).

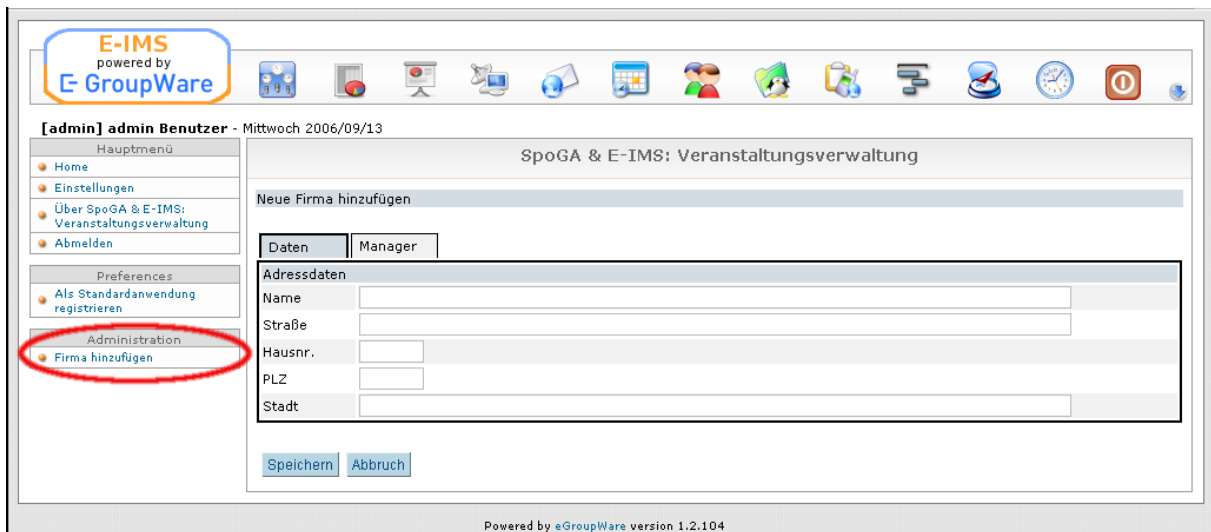


Abbildung 23: Hinzufügen neuer Firmen

7.1.10. Optionen des Firmenverantwortlichen

Der Firmenverantwortliche verfügt über Funktionen zum Hinzufügen neuer Benutzer zur Firmengruppe sowie bereits in der Firmengruppe vorhandene Nutzer als „Veranstalter“ zu registrieren, um diesen so die Möglichkeit zu gewähren, selbst Veranstaltungen anzulegen.

Hierzu existieren in der Veranstaltungsverwaltung die beiden Punkte „Mitglieder zur Firmengruppe hinzufügen“ sowie „Nutzer als Veranstalter registrieren“, die nur für den Firmenverantwortlichen sichtbar sind.

7.2. Das Modul „Abrechnung“

Dieses Modul dient der Abrechnung der im Kontext einer Veranstaltung genutzten Ressourcen. Veranstaltungsteilnehmer können hier eine Rechnung der von ihnen beanspruchten Ressourcen einsehen während dem Veranstalter eine Übersicht über sämtliche, im Zusammenhang mit der Veranstaltung angefallenen, Kosten zur Verfügung steht.

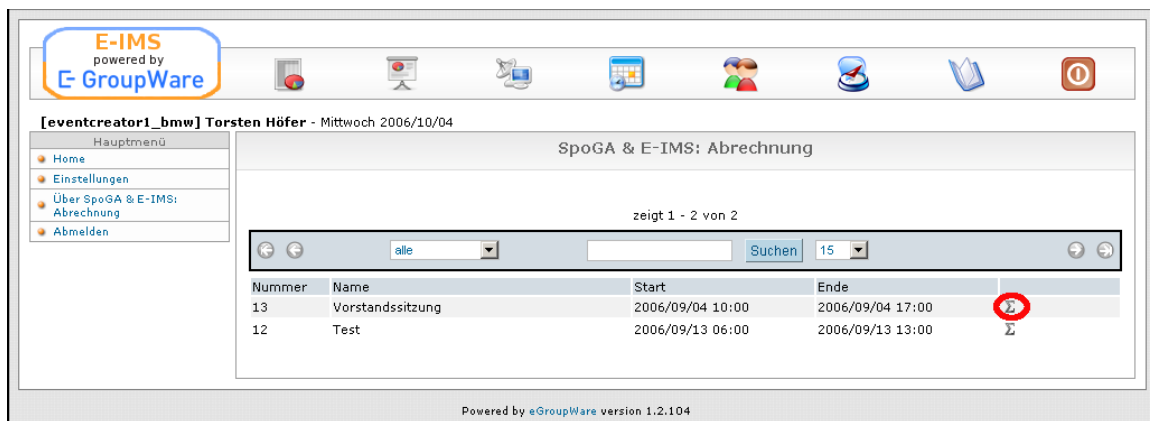


Abbildung 24: Veranstaltungsübersicht im Abrechnungsmodul

Hierzu werden die Veranstaltungen analog zum Modul „Veranstaltungsverwaltung“ in einer Listenansicht dargestellt (siehe Abbildung 24). Diese Listenansicht kann ebenfalls nach den im vorigen Kapitel beschriebenen Kriterien „alle Veranstaltungen“, „Veranstaltungsteilnahmen“ sowie „angelegte Veranstaltungen“ gefiltert werden. Auch die Suchfunktion sowie das Einschränken der Anzahl der angezeigten Veranstaltungen auf einen bestimmten Maximalwert ist ebenso möglich.

Durch Anklicken des Summenzeichens Σ (siehe Abbildung 24) gelangt man zur jeweiligen Kostenübersicht einer Veranstaltung. Diese stellt die gebuchten Ressourcen einer Veranstaltung inklusive eventuell angefallener SpoGA-Nutzungsgebühren in tabellarischer Form dar. Dabei sind für den Veranstalter die gebuchten Ressourcen jedes Veranstaltungsteilnehmers sichtbar, für den einzelnen Teilnehmer im Gegensatz dazu jedoch nur die von ihm selbst gebuchten Ressourcen. Mittels der Filteroptionen „Alle Rechnungen“ bzw. „Eigene Rechnung“ kann der Veranstalter auswählen, ob eine Komplettrechnung der Veranstaltung, d.h. eine Rechnung über die Ressourcen aller Teilnehmer, oder nur eine eigene Rechnung, d.h. eine Rechnung über die ausschließlich von ihm gebuchten Ressourcen, dargestellt werden soll (siehe Abbildung 25).

Angezeigt werden Informationen wie der Name des Nutzers, die Bezeichnung der gebuchten Ressource, die Nutzungsdauer, der Nutzungspreis pro Stunde sowie die, aus dem Produkt von Nutzungsdauer und Kosten pro Stunde resultierenden, Nutzungskosten. Die letzte Zeile der Tabelle enthält die aus der Summe der Nutzungsentgelte sämtlicher gebuchter Ressourcen errechneten Gesamtkosten.

Nummer	Nutzername	Ressource	Dauer	Preis/Einheit	Gesamtkosten
1	Hans Meier	Geha top vision 1400	1 Stunden	5.00	5.00
2	Sebastian Stiegler	Samsung SP-P300ME	4 Stunden	10.00	40.00
Summe					45.00

Abbildung 25: Rechnungsübersicht

Um die Daten mit externen Anwendungen weiterverarbeiten zu können enthält die Rechnungsübersicht eine Exportfunktion in das weit verbreitete CSV-Format. Die zu exportierenden Daten werden dabei als durch Komma getrennte Werte (comma separated values) in eine Textdatei geschrieben und stehen so z.B. Tabellenkalkulationsprogrammen zur Verfügung.

Um Abrechnungsdaten zu speichern genügt ein Klick auf den am Ende der Rechnungsübersicht dargestellten Button [CSV Export](#). Über den „Speichern“-Dialog des Browsers kann die generierte Datei an einem beliebigen Ort gespeichert werden. Neben seinen eigenen Rechnungsdaten kann der Veranstalter die Daten aller Veranstaltungsteilnehmer exportieren, was durch Betätigen des Buttons [CSV Export \(alle\)](#) geschieht.

8. Entfernter Zugriff via Mobiltelefon

8.1. Das Szenario

Das eGroupWare-System bietet eine Reihe von Schnittstellen zum Zugriff an (vergleiche Abbildung 6). Hierzu zählen neben dem Webinterface auch Funktionen zum Abgleich von Daten über das SyncML-Protokoll mit einem dieses unterstützenden Gerät oder der Zugriff via Webservice-Client über die beiden Protokolle SOAP sowie XML-RPC (vergl. [Cer02]).

Letzteres wurde im Rahmen dieser Studienarbeit dazu benutzt, um einem Veranstaltungsteilnehmer den mobilen Zugriff auf Teile des eGW-System zu gewähren. Hierzu wurde prototypisch eine Anwendung entwickelt, mit der die Daten einer Veranstaltung auf ein Mobiltelefon abgerufen und angezeigt werden können. Somit sind diese unabhängig vom Internetzugang über einen PC ständig abrufbar.

8.2. Eingesetzte Techniken und Werkzeuge

Als Zielplattform für die Entwicklung des Prototypen wurden Mobiltelefone mit dem Betriebssystem „Windows Mobile“¹⁸ ab der Version „Windows Mobile 2003“ ausgewählt. Betrachtet man die aktuelle Marktsituation der Betriebssysteme für mobile Endgeräte ist diese Entscheidung sicherlich in Frage zu stellen, da der eindeutige Marktführer in diesem Bereich das von Symbian entwickelte „Symbian OS“ mit einem geschätzten Marktanteil von über 70% darstellen dürfte (siehe hierzu auch [Sym]). Mit der „Java Platform Micro Edition (J2ME)“ existiert zudem eine inzwischen häufig genutzte Alternative, die die Entwicklung von plattformunabhängigen Anwendungen für mobile Endgeräte ermöglicht.

Die Entscheidung des Zielsystems zugunsten von Windows Mobile fiel jedoch schlussendlich aufgrund der vergleichsweise einfachen Programmierung von Anwendungen mit Hilfe von Microsoft Visual Studio 2005¹⁹ und der dazu einsetzbaren Programmiersprache C#. Der in Visual Studio integrierte Mobiltelefon-Emulator ermöglicht zudem ein einfaches Testen der entwickelten Applikation unter realitätsnahen Bedingungen.

Zur Realisierung des Zugriffs auf das eGW-System steht neben dem XML-RPC-Protokoll auch das heute häufig zur Realisierung von Webservices im Rahmen der dienstorientierten Architekturen (SOA) zum Einsatz kommende SOAP-Protokoll zur Auswahl. Die in der eGroupWare enthaltene Implementierung war zum Zeitpunkt der Entwicklung der Applikation jedoch noch nicht besonders ausgereift. Des Weiteren war die zugehörige Dokumentation unzureichend, so dass die Designentscheidung bezüglich des Remote-Zugriffs zugunsten des XML-RPC-Protokolls ausfiel. Dessen Funktionsweise wird im nachfolgenden Abschnitt 8.3 genauer erläutert.

¹⁸Windows Mobile: <http://www.microsoft.com/windowsmobile>

¹⁹Microsoft Visual Studio: <http://www.microsoft.com/germany/msdn/vstudio>

Während Bibliotheken für die Nutzung von SOAP bereits zur Standardausstattung des zur Entwicklung benutzten .NET-Frameworks gehören ist für die Nutzung von XML-RPC eine entsprechende externe Bibliothek zu verwenden. Zum Einsatz kommt hier die von CookComputing zur Verfügung gestellte Bibliothek „XML-RPC.NET“, die auch das von mobilen Endgeräten verwendete „.NET portable Framework“ unterstützt (siehe auch [Coo]).

8.3. Arbeitsweise von XML-RPC

Mit Hilfe des XML-RPC Protokolls können Methoden einer sich auf einem entfernten Rechner befindlichen Anwendung aufgerufen werden (vergl. [Cer02]). Diese so genannten Remote-Procedure-Calls werden heutzutage häufig verwendet, um eine Client-Server-Kommunikation zu ermöglichen. Herkömmliche Techniken waren dabei jedoch immer von einer bestimmten Implementierung und damit Programmiersprache abhängig, so mussten Client- und Server-Applikation in der gleichen Programmiersprache implementiert worden sein. Zur Beseitigung dieser Einschränkung verwendet XML-RPC die standardisierte Extensible Markup Language (XML) zum Datenaustausch zwischen den kommunizierenden Anwendungen. Sowohl der Methodenaufruf als auch die Rückgabe von Werten erfolgen in Form einer solchen XML-Nachricht. Damit ist es nun Client-Server-Anwendungen möglich, unabhängig der Implementierung miteinander zu kommunizieren. In der Regel kommen dabei so genannte Proxy-Klassen zum Einsatz, die das „Verpacken“ des Methodenaufrufes bzw. des Rückgabewertes in das XML-Format übernehmen. Analog dazu erfolgt auch das „Entpacken“ über solche Proxy-Klassen, die die XML-Nachricht auswerten und die entsprechende Aktion ausführen. Für viele heute am Markt befindlichen Programmiersprachen existieren Bibliotheken, die die entsprechenden Funktionalitäten bereitstellen.

Das Listings 6 enthält eine Anfrage an den E-IMS Server. Ausgeführt wird dabei ein Login am eGW-System durch Übermittlung eines Nutzernamens sowie eines, in diesem Falle leeren, Passwortes. Die ähnlich aussehende und daher hier nicht abgebildete Antwort enthält als Bestätigung eines korrekt ausgeführten Anmeldevorgangs eine bei nachfolgenden Anfragen zu verwendende „SessionID“ sowie einen weiteren, zur Authentifizierung benötigten, Parameter „KP3“.

Neben XML-RPC existiert mit SOAP ein weiteres Protokoll zur Realisierung von entfernten Methodenaufrufen. Auch hier geschieht die Kommunikation über XML als Austauschformat. Der Unterschied zwischen beiden Techniken liegt in der Komplexität: XML-RPC stellt nur eine kleine Menge der bei SOAP vorhandenen Datenformate und Funktionalitäten bereit. Die entsprechende Spezifikation (siehe [Win03]) umfasst lediglich sieben Seiten was in etwa einem Sechstel der vierzig Seiten umfassenden SOAP-Spezifikation (siehe [W3C07]) entspricht. Der Entwickler hat somit zu entscheiden, welches der beiden Protokolle den Anforderungen der zu entwickelnden Applikation am besten entspricht.

Listing 6: Aussehen einer XML-RPC Anfrage

```
<?xml version="1.0"?>
<methodCall>
  <methodName>system.login</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>domain</name>
            <value>
              <string>default</string>
            </value>
          </member>
          <member>
            <name>username</name>
            <value>
              <string>eventcreator1_bmw</string>
            </value>
          </member>
          <member>
            <name>password</name>
            <value>
              <string />
            </value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

8.4. Implementierung einer Beispielanwendung

Der entwickelte Prototyp bietet drei Funktionen, die das Ein- und Ausloggen am System sowie die Anzeige der Veranstaltungen des eingeloggtten Nutzers umfassen.

Zur Umsetzung war die Entwicklung von insgesamt sieben Klassen notwendig, welche im Klassendiagramm in Abbildung 26 ausführlich dargestellt sind und im Folgenden kurz beschrieben werden.

Die Klassen **Form1**, **Form2** und **Form3** sind für die Darstellung der grafischen Oberfläche zuständig, wobei jede Klasse eine Bildschirmmaske repräsentiert (vergleiche auch Abschnitt 8.5).

Die Verbindung zu einem entfernten Rechner, in diesem Fall dem eGW-System, wird durch die Klasse **Client** hergestellt, deren Instanz in der zweiten Bildschirmmaske, **Form2**, erzeugt wird. Dabei wird von der darin eingebetteten Proxy-Klasse **LoginProxy** Gebrauch gemacht. Diese ist Subklasse von **XmlRpcClientProtocol** aus der XML-RPC-

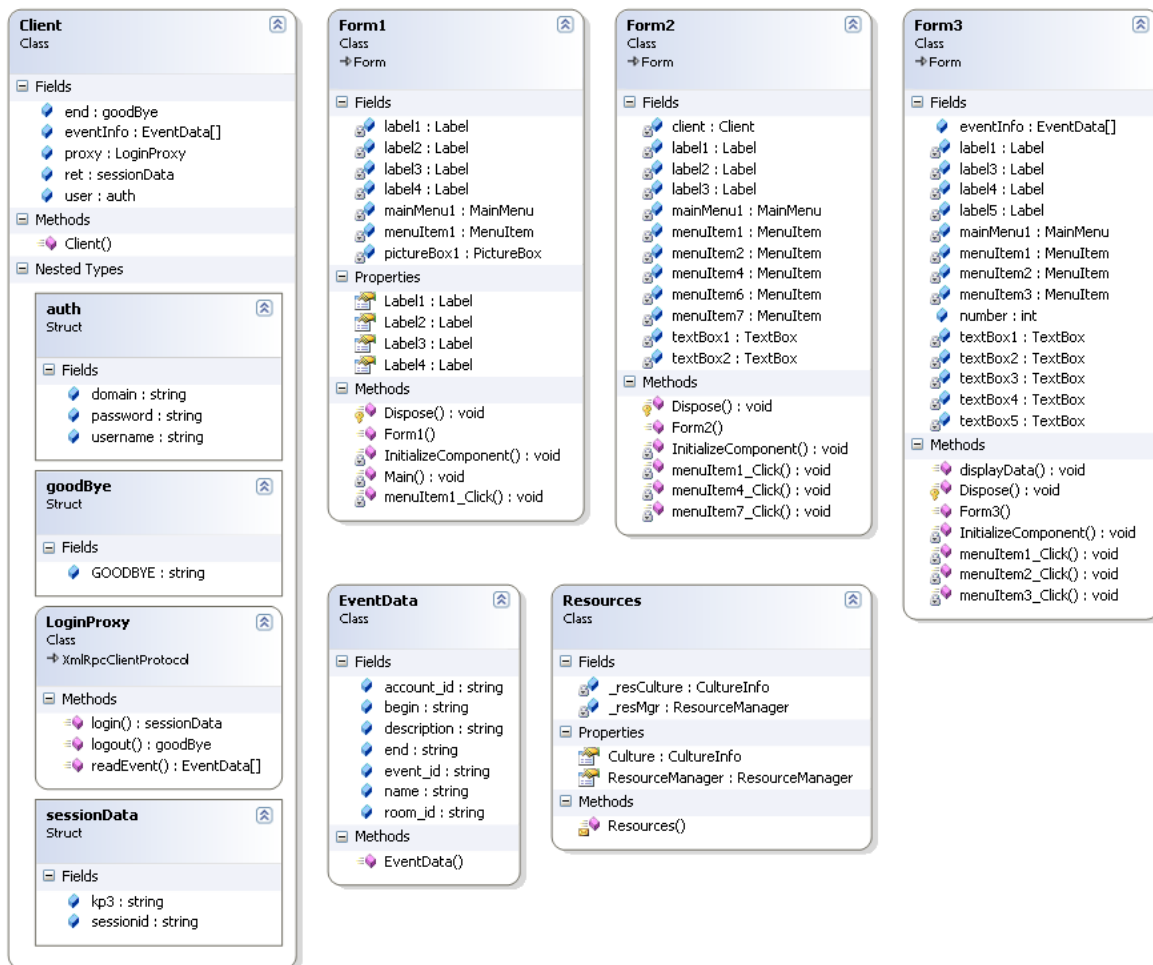


Abbildung 26: Klassendiagramm der Applikation zum entfernten Zugriff via Mobiltelefon

Bibliothek und somit in der Lage, die darin implementierten Funktionen zu nutzen. Daneben enthält sie die einen entfernten Methodenaufruf ausführenden Methoden `login()`, `logout()` sowie `readEvent()`. Erstere dient dem Login am System wobei als entfernte, vom eGroupWare-System bereitgestellte, Methode `system.login()` aufgerufen wird. Analog dazu erfolgt mit `logout()` das Abmelden vom System durch den Aufruf von `system.logout()` auf Seiten des eGW-Systems. Die Methode `readEvent()` realisiert die Abfrage der Veranstaltungen des gerade angemeldeten Nutzers, wozu die entfernte Methode `spoga_event_bo.read()` aufgerufen wird.

Mit der Klasse `EventData` wird der gleichnamige Datentyp verfügbar gemacht, der zur Speicherung der Informationen zu einer Veranstaltung dient und in Form eines Arrays den Rückgabewert der Methode `readEvent()` darstellt.

Anfänglich kam es bei der Auswertung der vom eGW-System gesendeten Antworten

zu Problemen, die zu einem „Abstürzen“ der Anwendung führten. Mit Hilfe des Programmes „TcpTrace“²⁰, das ein Protokollieren des Netzwerkverkehrs ermöglicht, konnte die Ursache ermittelt werden: Die vom eGW-System gesendete XML-Nachricht enthielt unerlaubte Leerzeichen, deren Vorhandensein der Parser der XML-RPC.NET-Bibliothek durch Ausgabe einer Fehlermeldung quittierte. Da die Ursache der Existenz dieser Leerzeichen nicht ausfindig zu machen war und auch die Lösungsvorschläge der äußerst hilfsbereiten Entwicklergemeinde des eGroupWare-Systems keinerlei Abhilfe brachten konnte letztendlich nur das Abändern des Parsers der XML-RPC.NET- Bibliothek diesen Fehler beseitigen. Charles Cook, der Hauptentwickler der Bibliothek, ergänzte diese auf meine Nachfrage hin um eine Option, die den Parser tolerant für diese eigentlich nicht konformen Leerzeichen macht. Mit der so durchgeführten Modifikation gelingt nun der problemlose Zugriff per XML-RPC auf das eGW-System.

8.5. Anleitung zur Nutzung der Mobiltelefonanwendung

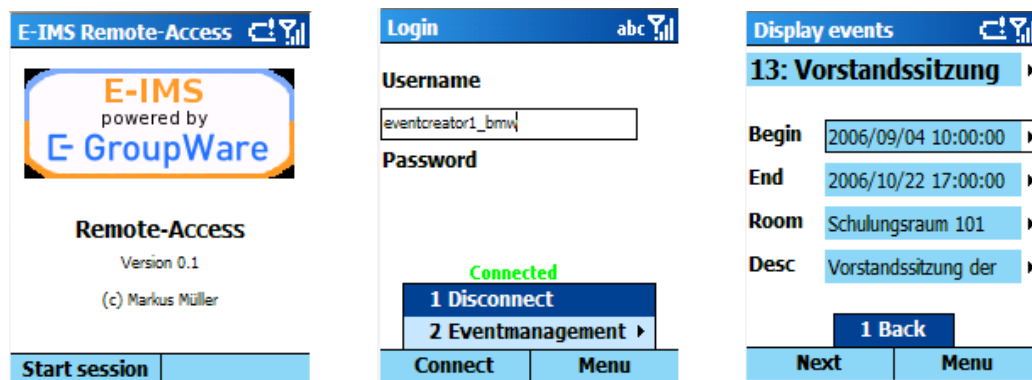


Abbildung 27: (a) Begrüßungsbildschirm (b) Login-Fenster (c) Anzeige von Veranstaltungen

Neben der Abfrage der Veranstaltungsdaten eines Nutzers über die im vorigen Kapitel vorgestellte Weboberfläche existiert auch die Möglichkeit einer Abfrage mittels eines Windows Mobile 2003 kompatiblen Mobiltelefons.

Die Benutzung der entsprechenden Softwarekomponente gestaltet sich dabei relativ einfach. Nach dem Starten erscheint der in Abbildung 27(a) dargestellte Begrüßungsbildschirm. Durch Auswählen von „Start Session“ gelangt man zur nächsten Bildschirmmaske (Abbildung 27(b)).

In dieser sind die Anmeldeinformationen wie Nutzernamen und das zugehörige Passwort einzutragen. Durch Auswählen von „Connect“ wird der Anmeldevorgang am eGW-System gestartet. Im Falle einer erfolgreichen Anmeldung erscheint in grüner Schrift die

²⁰TcpTrace: <http://www.pocketsoap.com/tcptrace>

Mitteilung „Connected“, andernfalls die Meldung „An error has occurred! Your username/password is not valid!“. Nach dem erfolgreichen Anmelden stehen über die Schaltfläche „Menu“ zwei Optionen zur Verfügung, wobei erstere das Ausloggen vom System und die zweite das Anzeigen der Veranstaltungen ermöglicht.

Beim erfolgreichen Ausloggen erfolgt eine Rückkehr zum Begrüßungsbildschirm.

Die Anzeige der Veranstaltungen hingegen erfolgt in einer neuen Bildschirmmaske (Abbildung 27(c)), bei der mittels Button „Next“ die nächste vorhandene Veranstaltung angezeigt werden kann. Ein Zurückspringen zur vorigen Veranstaltung ist zum jetzigen Zeitpunkt noch nicht möglich. Wird die letzte Veranstaltung angezeigt beginnt die Anzeige nach dem Betätigen des Buttons „Next“ wieder mit der ersten Veranstaltung. Über den rechten Button „Back“ gelangt man zur vorigen Bildschirmmaske zurück.

9. Abschließende Betrachtungen

9.1. Mögliche Erweiterungen

Erweiterungen sind insbesondere im Bereich der dem E-IMS-Administrator zur Verfügung stehenden Funktionalitäten denkbar.

Bei der Registrierung einer Firma am System über die dafür bereitgestellte Funktion (vergl. 7.1.9) werden automatisiert die entsprechenden Nutzergruppen sowie einige Ressourcenkategorien erzeugt. Eine Rechtevergabe zur Nutzung dieser Ressourcenkategorien erfolgt jedoch aktuell noch nicht und muss manuell vom E-IMS-Administrator vorgenommen werden.

Bisher können Firmenkonto zwar angelegt werden, ein Ändern der Daten oder ein Entfernen eines solchen Kontos ist jedoch noch nicht möglich und muss manuell durch den E-IMS-Administrator über die von der Groupware zur Verfügung gestellten Funktionen erfolgen. Ähnlich wie das Anlegen eines Firmenkontos sollen diese Einzelschritte in einer einzigen Funktion zusammengefasst werden, um so die Nutzung zu vereinfachen.

Mit der Umsetzung eines bestimmten Geschäftsmodells (vergleiche dazu Kapitel 3.5) müsste dem E-IMS-Administrator die Möglichkeit der Abrechnung der Nutzung des Systems gegeben werden.

Im Bereich der Mobilfunkanwendung könnten weitere Funktionalitäten realisiert werden. Ist bisher nur das Abfragen der Veranstaltungsdaten möglich, könnte in Zukunft beispielsweise auch das Buchen von Ressourcen hierüber erfolgen.

9.2. Fazit

Mit den beiden im Rahmen dieser Studienarbeit entwickelten Modulen wurde die Grundlage für das E-IMS geschaffen. Dessen Funktionalitäten ermöglichen das einfache Organisieren von Veranstaltungen aller Art unter Minimierung des damit verbundenen Verwaltungsaufwands. Im Zusammenspiel mit dem SpoGA-System als Ergebnis der Studienarbeit von Matthias Ehrenstein erlaubt das Gesamtsystem auch das automatisierte Freischalten des Zugangs der Netzwerkgeräte der Veranstaltungsteilnehmer zu einem (Firmen-) Netzwerk, wovon neben dem Organisator einer Veranstaltung auch die Netzwerkadministratoren profitieren.

Mit dem Prototyp der Applikation zum mobilen Zugriff auf das E-IMS wurde die Vielseitigkeit des verwendeten Basissystems demonstriert und eine Grundlage für Erweiterungen geschaffen.

A. Interview zum geplanten Einsatz eines Groupware-Systems an der Universität Koblenz-Landau

Über die Seiten des Rechenzentrums der Universität Koblenz-Landau wurden wir auf einen aktuell statt findenden Test von Groupware-Systemen an der Universität aufmerksam. Um von den im Rahmen dieses Tests gesammelten Erfahrungen profitieren zu können, vereinbarten wir mit dem Wissenschaftlichen Leiter des Rechenzentrums einen Gesprächstermin. Im Folgenden findet sich eine Zusammenfassung dieses Gespräches, welches am 13.03.2006 um 10 Uhr stattfand.

Frage: Welche Gründe gab es für die Auswahl der beiden Systeme Open-Xchange [OX] und OpenGroupware [Ope] ?

Herr Arndt:

Das Hauptargument für den Einsatz der beiden oben genannten Systeme ist ihre Verfügbarkeit als OpenSource-Produkte. Open-Xchange wurde bereits früher in der zu diesem Zeitpunkt kommerziellen Version SUSE Mail 4 getestet. Für einen Produktiveinsatz als Groupware-Lösung wäre jedoch der kostenintensive Erwerb von Lizenzen notwendig gewesen, so dass diese Möglichkeit nicht weiter verfolgt wurde.

Frage: Welche Anforderungen stellt die Universität an die Groupware-Systeme?

Herr Arndt:

Da bereits ein Großteil der Funktionen einer typischen Groupware-Lösung von anderen, bereits im Einsatz befindlichen Systemen wie dem BSCW-Server²¹ abgedeckt wird, war die einzige wirkliche Hauptanforderung an das einzusetzende System die Bereitstellung eines Gruppenkalenders zur gemeinsamen Terminplanung der Teilnehmer eines Projektes.

Frage: Welche Erfahrungen haben Sie bisher mit den beiden Systemen gemacht?

Herr Arndt:

Zwischen den beiden Produkten existieren bezüglich der Funktionalitäten nur minimale Unterschiede. Aufgrund der nutzerfreundlicheren grafischen Oberfläche gibt es aber eine Präferenz für Open-XChange. Bisher wurde die Software nur mit einer geringen Anzahl von Nutzern getestet, daher steht noch der Test unter realen Bedingungen, also insbesondere einer hohen Anzahl an Zugriffen, aus.

²¹BSCW-Server: <http://bscw.fit.fraunhofer.de>

Frage: Traten beim Probetrieb der Systeme besondere Fehler auf?

Herr Arndt:

Gelegentlich kam es bei Open-XChange zu Problemen bei der Synchronisation mit Microsoft Outlook. Bei OpenGroupware hingegen sind keine erwähnenswerten Fehlfunktionen aufgetreten.

Frage: Welches der beiden Systeme soll in Zukunft produktiv eingesetzt werden?

Herr Arndt:

Geplant ist die Nutzung von Open-Xchange in der Universität Koblenz ab dem Sommersemester 2006. OpenGroupware wird nicht zum Einsatz kommen, da uns die Gestaltung der Oberfläche nicht zusagte.

Literatur

- [Aya] AYALA, Dietrich: *Anleitung zur Entwicklung von Anwendungen mit Hilfe der NuSOAP-Bibliothek*. <http://dietrich.ganx4.com/nusoap/>, Abruf: 28.08.2006
- [Cer02] CERAMI, Ethan: *Web Services Essentials - Distributed Applications with XML-RPC, SOAP, UDDI & WSDL*. First Edition. O'Reilly, 2002. – ISBN 0-596-00224-6
- [Coo] COOK, Charles: *Anleitung zur Entwicklung von Anwendungen mit Hilfe der XML-RPC-Bibliothek*. <http://www.xml-rpc.net/faq/xmlrpcnetfaq.html>, Abruf: 28.08.2006
- [EGK06] EBERSBACH, Anja ; GLASER, Markus ; KUBANI, Radovan: *Joomla! Das Handbuch für Einsteiger*. Erste Auflage. Galileo Press, 2006
- [eGr] EGROUPWARE: *Anleitung zur Entwicklung von Modulen für das eGroupWare-System*. http://www.egroupware.org/index.php?page_name=wiki&wikipage=DeveloperDocs, Abruf: 28.08.2006
- [Ehr07] EHRENSTEIN, Matthias: *SpoGA - Spontaneous WLAN Guest Access*. April 2007. – Studienarbeit, Universität Koblenz-Landau.
- [Ell06] ELLER, Frank: *Visual C# 2005 Grundlagen und Programmier Techniken*. Zweite Auflage. Addison-Wesley, 2006. – ISBN 382732288X
- [Eur06] EUROSTAT: *Internet-Nutzung in der EU25 - Pressemitteilung 146/2006*. Version: November 2006. http://epp.eurostat.ec.europa.eu/pls/portal/docs/PAGE/PGP_PRD_CAT_PREREL/PGE_CAT_PREREL_YEAR_2006/PGE_CAT_PREREL_YEAR_2006_MONTH_11/4-10112006-DE-AP.PDF, Abruf: 31.04.2007
- [Fas] FASTTEMPLATES: *Homepage der Template-Engine FastTemplates*. <http://www.thewebmasters.net/>, Abruf: 28.08.2006
- [GK07] GROSS, Tom ; KOCH, Michael: *Computer-Supported Cooperative Work*. Michael Herczeg, 2007. – ISBN 3486580000
- [Kan04] KANNENGIESSER, Matthias: *PHP5/MySQL 4*. Erste Auflage. Franzis, 2004. – ISBN 3772370055
- [KE04] KEMPER, Alfons ; EICKLER, Andre: *Datenbanksysteme- Eine Einführung*. Fünfte Auflage. Oldenbourg, 2004. – ISBN 3486273922

-
- [Moz] MOZILLA: *Homepage der XML User Interface Language (XUL)*. <http://www.mozilla.org/projects/xul/>, Abruf: 28.08.2006
- [MyS] MYSQL: *Handbuch des Datenbanksystems MySQL*. <http://www.mysql.org/doc/refman/5.0/en/index.html>, Abruf: 28.08.2006
- [Ope] OPENGROUWARE: *Homepage des Groupware-Systems OpenGroupware*. <http://www.opengroupware.org/>, Abruf: 28.08.2006
- [Orr] ORR, David: *Templates, The PHPLIB Way*. <http://www.phpbuilder.com/columns/david20000512.php3>, Abruf: 28.08.2006
- [OX] OPEN-XCHANGE: *Homepage des Groupware-Systems Open-Xchange*. <http://www.open-xchange.com/EN/>, Abruf: 28.08.2006
- [PHPa] PHP: *Handbuch der Skriptsprache PHP*. <http://www.php.net/manual/de/>, Abruf: 28.08.2006
- [PHPb] PHPLIB: *Homepage der PHPBaseLibrary (PHPLib)*. <http://phplib.sourceforge.net>, Abruf: 28.08.2006
- [Sch] SCHUMANN, Sascha: *Templates- Why and How To Use Them In PHP3*. <http://www.phpbuilder.com/columns/sascha19990316.php3>, Abruf: 28.08.2006
- [Sha] SHAFRANOVICH, Y.: *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. <http://tools.ietf.org/html/rfc4180#page-2>, Abruf: 28.08.2006
- [sma] SMARTY: *Homepage der Template-Engine smarty*. <http://smarty.php.net/>, Abruf: 28.08.2006
- [Spe07] SPEICH, Christoph: *Spontaneous Guest Access (SpoGA) & Extended Invitation Management System (E-IMS): Modul SpoGA Manager und Erweiterung der Ressourcenverwaltung*. Juni 2007. – Studienarbeit, Universität Koblenz-Landau.
- [Sym] SYMBIAN: *Fast facts*. <http://www.symbian.com/about/fastfacts/fastfacts.html>, Abruf: 28.08.2006
- [Tro05] TROELSEN, Andrew: *Pro C# 2005 and the .NET 2.0 Platform*. Third Edition. Apress, 2005. – ISBN 1590594193
- [W3C07] W3C: *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)*. Version: April 2007. <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/>, Abruf: 12.06.2007

-
- [War02] WARM, Markus: *3-Tier Architektur*. Version: Februar 2002. <http://mwvb.de/3tier01.htm>, Abruf: 28.08.2006
- [Win03] WINER, Dave: *XML-RPC Specification*. Version: Juni 2003. <http://www.xmlrpc.com/spec>, Abruf: 12.06.2007
- [WT05] WELLING, Luke ; THOMSON, Laura: *PHP 5 und MySQL 5 Kompendium. Dynamische Webanwendungen von Einstieg bis E-Commerce*. Markt und Technik, 2005. – ISBN 3827268095