

Pflichtenheft

SEDA-WI-Proj-B

SF GmbH

Denis Hamann Matr.-Nr. 1684873

Anna Kupfer Matr.-Nr. 141515

Hannes Stadler Matr.-Nr. ?

Christian Hindelang Matr.-Nr. ?

Otto-Friedrich-Universität Bamberg

Version: v0.1 - 28. Oktober 2012

Inhaltsverzeichnis

1	Preamble	2
2	Preamble	3
3	Preamble	4
4	Produktumgebung	5
4.1	Software	5
4.2	Hardware	5
4.3	Orgware	5
4.4	Produkt-Schnittstellen	5
5	Funktionen	6
5.1	Prozesse mit Dateneingabe	6
5.2	Listen	9
5.3	Berichte	10
6	Daten	11
7	Leistungen	13

Abbildungsverzeichnis

1	SERM der Daten-Architektur	11
---	--------------------------------------	----

1 Preamble

We'd like to address the reader of this documentation before going into the details of this documentation that the project itself encountered several challenges which couldn't all be solved in a satisfying way. These lead to a program with a limited functionality and a documentation with some major gaps. For reasons and the progress itself which lead to this situation please refer to chapter 8.

2 Preamble

We'd like to address the reader of this documentation before going into the details of this documentation that the project itself encountered several challenges which couldn't all be solved in a satisfying way. These lead to a program with a limited functionality and a documentation with some major gaps. For reasons and the progress itself which lead to this situation please refer to chapter 8.

3 Preamble

We'd like to address the reader of this documentation before going into the details of this documentation that the project itself encountered several challenges which couldn't all be solved in a satisfying way. These lead to a program with a limited functionality and a documentation with some major gaps. For reasons and the progress itself which lead to this situation please refer to chapter 8.

4 Produktumgebung

Im folgenden wird die Umgebung beschrieben, in der die spätere Software eingesetzt werden wird.

4.1 Software

Betriebssystem: Windows 7 (32/64bit) Laufzeitsystem: Java JRE 7.0 Datenbank: PostgreSQL 9.2.1

Shell: Windows Shell (explorer.exe) Fensterprogramm, kein CLI Programm

4.2 Hardware

CPU: Intel x86/x65 Architektur - >1Ghz >=1GB DDR2RAM HDD: >100MB Maus & Tastatur Netzwerkanschluss (für DB Anbindung)

4.3 Orgware

Eine entsprechende Netzwerkinfrastruktur muss vorhanden sein um eine Verbindung zur Datenbank herstellen zu können. Die Datenbank (Postgresql) muss auf einem entsprechenden Server installiert sein und als Dienst laufen. Es muss sicher gestellt werden, dass entsprechende Router und Firewalls konfiguriert sind und eine Verbindung in beide Richtungen ermöglichen. Es wird davon ausgegangen, dass die Hausverwaltung, sowie die Dozenten entsprechende Informationen zur Räumen und Lehrveranstaltungen korrekt und zeitnah in das System einpflegen um eine sinnvolle Nutzung zu ermöglichen.

4.4 Produkt-Schnittstellen

5 Funktionen

5.1 Prozesse mit Dateneingabe

/F01/ **Prozess mit Dateneingabe:** Benutzer-Authentifizierung

Akteur: Alle

Beschreibung: Alle Benutzer des Systems haben die Möglichkeit sich mittels einer Login-Funktion am System anzumelden. Hierdurch werden Sie einer der möglichen Benutzergruppen zugeordnet. Nicht angemeldete Benutzer gelten als Mitglieder der Benutzergruppe SStudenten".

Eingabevalidierung: Nutzer müssen eine gültige Nutzerkennung mit dazu passendem Passwort eingeben. Diese werden von der Verwaltung festgelegt. Wünschenswert wäre, dass Studenten sich auch am System anmelden können (siehe /FW61/) und hierzu ihre normale Hochschulkennung und Passwort verwenden könnten. Im Falle von falschen Zugangsdaten erhält der Benutzer eine Meldung, die ihn über den Fehlversuch informiert. Es gibt nur eine Meldung, so dass für ihn nicht ersichtlich ist, ob das Passwort falsch war oder er schon eine ungültige Nutzerkennung eingegeben hat.

/F20/ **Prozess mit Dateneingabe:** Verwalten von Lehrveranstaltungen

Akteur: Dozenten

Beschreibung: Dozenten können Lehrveranstaltungen, die sie oder andere Mitarbeiter ihres Lehrstuhls in einem Semester halten eintragen, bearbeiten und löschen. (Diese sind nur für angehörige ihres Lehrstuhls sichtbar und bearbeitbar. Für Studenten werden sie erst sichtbar und verwendbar, wenn für sie ein Raum gebucht wurde und dies bestätigt wurde.)

Eingabevalidierung: Es können beliebige Lehrveranstaltungen erstellt, gelöscht oder bearbeitet werden. Doppelerstellungen sind möglich, falls dies in speziellen Fällen von Dozenten benötigt wird (durch die systeminterne ID ist ein fehlerfreier Systembetrieb gewährleistet). Die übrigen Attribute einer Lehrveranstaltung (/D50/) werden sofern möglich nach Logik und passender Größe überprüft und bei fehlerhaften Eingaben wird ein Bearbeitungs- oder Erstellvorgang abgebrochen, sowie der Nutzer darauf hingewiesen.

/F10/ **Prozess mit**

Akteur: Ver

Beschreibung:

len, bearbeit

Eingabevalidierung:

oder bearbeit

(siehe Datene

(die systemin

so wie das au

wird der Nut

übrigen Attrib

Logik und pa

ben wird ein

der Nutzer d

weise nur ein

/FW21/ **Prozess mit**

gen

Akteur: Doz

Beschreibung:

Meldungen er

kommen. Es

sichtbar ist. V

fil haben und

einer Lehrver

zeigt werden.

Eingabevalidierung:

erstellt, gelö

bute einer Ti

Logik und pa

ben wird ein

der Nutzer d

/F30/ Prozess mit Dateneingabe: Buchen von Räumen**Akteur:** Dozenten

Beschreibung: Dozenten können ihre Lehrveranstaltung für ein Semester zu einer bestimmten Zeit an einem bestimmten Werktag in einem Raum eintragen. Eine Zeiteinheit beträgt dabei 2 Stunden, kann immer nur voll belegt werden und die Zählung der Einheiten geht von 8 Uhr morgens bis 20 Uhr abends (womit ein Tag aus 6 Einheiten besteht). Buchungen müssen seitens der Verwaltung bestätigt werden (siehe /F40/).

Eingabevalidierung: Es sind Buchungen für bereits eingetragene Lehrveranstaltungen möglich. Eine Buchung umfasst genau eine Zeiteinheit zu einem Raum an einem Tag. Es sind beliebig viele Buchungen für eine Lehrveranstaltung möglich. Eine Buchung darf sich nicht mit einer bereits von der Verwaltung bestätigten Buchung im gleichen Raum zur gleichen Zeit am gleichen Tag überschneiden, sonst wird der Vorgang abgebrochen und dem Nutzer ein entsprechender Fehler gezeigt. Die übrigen Attribute einer Buchung (/D60/) werden sofern sie vom Dozenten anzugeben sind und möglich sind nach Logik und passender Größe überprüft und bei fehlerhaften Eingaben wird ein Bearbeitungs- oder Erstellvorgang abgebrochen, sowie der Dozent darauf hingewiesen.

/F40/ Prozess mit

inklusive Kon

Akteur: Ver**Beschreibung**

wünsche zu R

umlegen. Im

sie dem Doze

terungen hint

automatische

Eingabevalidi

/DW80/ spez

dass Falschein

kann zusätzli

Größe validie

wiesen und d

/F50/ Prozess mit Dateneingabe: Verwaltung von Nutzerdaten**Akteur:** Verwaltung**Beschreibung:** Mitglieder der Verwaltung können Benutzer (Dozenten und Verwaltungsmitglieder) in das System einpflegen, sie bearbeiten oder löschen.**Eingabevalidierung:** Das Eintragen und Bearbeiten umfasst die in /D30/ spezifizierten Attribute (außer dem letzten Login und dem Salt, welche automatisch vom System generiert werden und wie gehabt der systeminternen ID). E-Mail und Benutzerkennung dürfen nicht schon im System vorhanden sein, anderenfalls erscheint eine entsprechende Fehlermeldung und der Vorgang wird abgebrochen. Das Passwort wird beim Erstellen und Bearbeiten direkt zur Laufzeit zusammen mit einem benutzerspezifischen zufälligen Wert (dem SSalt") in einen SHA512-Hash umgewandelt und nur als solcher persistent gespeichert. Es muss mindestens 8 Zeichen umfassen. Alle Attribute werden sofern möglich nach Logik und passender Größe überprüft und bei fehlerhaften Eingaben wird ein Bearbeitungs- oder Erstellvorgang abgebrochen, sowie der Nutzer darauf hingewiesen.**/F51/ Prozess mit****Akteur:** Ver**Beschreibung:**

das System ei

Eingabevalidierung:

in /D20/ spe

bereits im Sy

sprechende F

Alle Attribut

der Größe ü

Bearbeitungs

darauf hingew

/F60/ Prozess mit Dateneingabe: Belegung von Lehrveranstaltungen**Akteur:** Studenten**Beschreibung:** Studenten können sich für Lehrveranstaltungen einschreiben (diese Einschreibungen werden nicht persistent im System gespeichert). Überschneidungen (Belegung zur gleichen Zeit) ist möglich, das System generiert aber einen Warnhinweis.**Eingabevalidierung:** Das System bietet nur die Möglichkeit eine öffentliche Veranstaltung als hinzuzufügen oder diese wieder zu entfernen. Eine vom Benutzer erstellte Eingabe und somit eine Validierung ist nicht von Nöten. Wünschenswert wäre die persistente Speicherung dieser Zuordnungen wie sie durch die Funktion /FW61/ möglich wäre und im Datenobjekt /DW70/ spezifiziert ist.**/FW61/ Prozess mit**

möglichkeiten

Akteur: Stud**Beschreibung:**

Profil von sic

mit welchem

Eingabevalidierung:

te E-Mail, Pa

für Studenten

ben ghasht u

Adresse darf

erscheint eine

abgebrochen.

nach Logik u

Eingaben wir

sowie der Nut

5.2 Listen

/F70/	<p>Liste: Lehrveranstaltungen eines Lehrstuhls</p> <p>Akteur: Dozenten</p> <p>Beschreibung: Dozenten können sich im System eine Liste von allen Lehrveranstaltungen, die ihrem Lehrstuhl zugeordnet sind anzeigen lassen. Hierbei steht ein Filter zur Verfügung, der nur bestimmte Semester oder bestimmte Dozenten anzeigt.</p> <p>Enthaltene Daten: Name der Veranstaltung, Dozent, Semester, Tag(e), Zeit(en), gebuchter Raum, Freigabestatus, <i>wünschenswert eingeschriebene Studenten</i></p>	/F80/	<p>Liste: Buchu</p> <p>Akteur: Ver</p> <p>Beschreibung: eine Liste mit nen oder abg steht ein Filt bestatus, Tag tete Raumbu an erster Stel</p> <p>Enthaltene Semester, Ta Verweis auf <i>schriebene St</i></p>
/F90/	<p>Liste: Nutzer</p> <p>Akteur: Verwaltung</p> <p>Beschreibung: Mitglieder der Verwaltung können sich im System eine Liste mit allen Benutzern des Systems anzeigen lassen. Es steht ein Filter zur Verfügung, der es erlaubt nach Vor- oder Nachnamen sowie der Nutzerzugehörigkeit zu filtern.</p> <p>Enthaltene Daten: Nutzerkennung, Vorname, Nachname, Nutzerzugehörigkeit (Lehrstuhl, Verwaltung oder ggf. auch Student), Zahl der Lehrveranstaltungen im aktuellen Semester, Datum des letzten Logins</p>	/F100/	<p>Liste: Lehrve</p> <p>Akteur: Stue</p> <p>Beschreibung: sich im System Lehrveranstal gung, der es Veranstaltung</p> <p>Enthaltene Semester, Tag <i>schriebene St</i></p>
/F110/	<p>Liste: Live-Ticker Meldungen</p> <p>Akteur: Studenten (sowie wünschenswerterweise Verwaltung und Dozenten mit Bearbeitungsfunktionen)</p> <p>Beschreibung: Studenten (aber auch anderen Nutzern) bekommen im System automatisch Live-Ticker Meldungen angezeigt, die die nächsten (falls möglich zu ihren Stundenplänen passenden) Lehrveranstaltungen anzeigen, sowie wünschenswerterweise auch von Dozenten oder Verwaltung zu diesen erstelle Meldungen.</p> <p>Enthaltene Daten: Lehrveranstaltungsname, Raum, Zeit oder Meldungstexte</p>		

5.3 Berichte

/F120/ **Bericht:** Raumbelegung

Akteur: Verwaltung (und wünschenswerterweise andere Nutzer)

Beschreibung: Mitglieder der Verwaltung (und wünschenswerterweise auch andere Nutzer) können sich eine Übersicht zur Belegung eines bestimmten Raumes in einem Semester anzeigen lassen, welche als Ansicht im System aufrufbar und als PDF exportierbar bzw. druckbar ist.

/F130/ **Bericht:** Semesterplan

Akteur: Studierende

Beschreibung: Die Belegungen in einem Semester werden hierbzw. als Stundenplan dargestellt und als PDF exportierbar bzw. druckbar ist.

Abbildung 1: SERM der Daten-Architektur

6 Daten

/D10/ **Datentyp:** Räume

Attribute: Raum ID (*systemintern*), Raumnummer, Gebäude, Stockwerk, Sitzplätze, PC-Plätze, Beamer, Visualizer, Overheads, Tafeln, Whiteboards

/D20/ **Datentyp:** L

Attribute: L
stuhlinhaber,

/D30/ **Datentyp:** Benutzer

Attribute: Benutzer ID (*systemintern*), Benutzerkennung, Passwort (Hash), Salt, E-Mail, Benutzerzugehörigkeit (Verwaltung, betreffender Lehrstuhl, wünschenswerterweise auch Student), Vorname, Nachname, letzter Login

/D50/ **Datentyp:** L

Attribute: V
zent), Veran
mester, Benö
be durch Doz
"Raumbelegu
trag erstellt v
mester buche

/D60/ **Datentyp:** Raumbelegungen (aller Freigabestatus-Arten)

Attribute: Belegungs ID (*systemintern*), Veranstaltungs ID (Lehrveranstaltung), Raum ID, Semester, Tag, Zeiteinheit, Freigabestatus (unbearbeitet|freigegeben|abgelehnt|gegenvorschlag), Freigabenachricht (Falls ein Vorschlag abgelehnt wurde und dies nun ein reservierter Vorschlag des Status "gegenvorschlagist), Kommentar

/DW70/ **Datentyp:** S

Attribute: S
(Student), Be

/DW80/ **Datentyp:** Ticker-Nachricht

Attribute: Meldungs ID (*systemintern*), Meldungstext, Start-Datum, End-Datum, exklusiv für Lehrstuhl ID ('null' wenn für alle), exklusiv für Veranstaltungs ID ('null' wenn für alle), exklusiv für Raum ID ('null' wenn für alle)

In Structured-Entity-Relationship-Modell (SERM) wird der Zusammenhang der oben spezifizierten persistent zu speichernden Daten verdeutlicht.

Wie zu sehen ist, existieren sowohl Räume als auch Lehrstühle als eigenständige Einheiten (Entity). Benutzer können einem Lehrstuhl zugeordnet werden, womit sie der Klasse "Dozententsprechen. Sie können aber auch der Klasse "Verwaltungöder SStudenten angehören und somit keinem Lehrstuhl zugeordnet werden. Lehrveranstaltungen müssen einem Benutzer zugeordnet werden, der (allein an der Grafik nicht erkennbar) ein der Klasse "Dozenten angehören muss. Eine Raumbelegung ist keine

Einheit sondern eine Zuordnung (Relationship) und muss einer Lehrveranstaltung und einem Raum zugeordnet werden können. Die Zuordnung Studentenbelegung muss einem Benutzer (der Klasse SStudent") und einer Raumbelegung zugeordnet werden können. Ticker-Nachrichten können einem Raum, einer Lehrveranstaltung oder einem Lehrstuhl zugeordnet werden, nichts davon ist aber zwingend.

7 Leistungen

/L10/ Alle Daten, die im vorherigen Abschnitt aufgelistet sind, müssen, sofern sie realisiert werden, persistent mittels einer SQL-Datenbank (erreichbar und verwaltet durch einen PostgreSQL-Server) gespeichert werden.

/L20/ Bei allen Pro
aussagekräftig
gemäß durchg

/L40/ Die Programm
bei eine saub
leistet wird,
Dieses erlaub
einem spätere
Die erste Ver
stehen:

/L30/ Die Realisierung erfolgt als Java-Anwendung, so dass ein möglicher Betrieb auf allen gängigen PCs der Universität sichergestellt ist.

CoreGUI: H
der Anwendu
von diesem M
Verwaltung
gesetzt, die d
/F40/, /F50/
Verwaltung
gesetzt, die d
/F40/, /F50/

/L50/ Es wird eine Nutzerauthentifizierung realisiert, die es gewährleistet, dass nur berechtigte Nutzer Änderungen durchführen. Um Missbrauch ausschließen zu können, sollte allerdings eine Middleware (die großteils die Model- und Controller-Schicht abdeckt) zwischen Datenbank und Client (der hauptsächlich die View-Schicht umsetzt) realisiert werden. In diesem Fall sollte die Kommunikation zwischen Client und Middleware durch SSL (mithilfe von javax.net.ssl.* und javax.rmi.ssl.*) abgesichert werden.

/L60/ Zeitliche Spez