

Teilentwurf

LISE E-Learning System

Matthias Englert, Fabian Schilha, Andreas Rottach

Wintersemester 2014/2015

Inhaltsverzeichnis

1	Datenbankentwurf	3
1.1	Datebankdiagramm	3
1.2	Beschreibung der Tabellen	4
2	System-Architektur	5
2.1	Kommunikationsdiagramme	5
2.2	Klassendiagramm	7
2.3	Methodenbeschreibung	8
2.3.1	Controller Methoden	8
2.3.2	Benutzerverwaltung Methoden	9
2.3.3	Datenbankmanager Methoden	10
2.3.4	RegistrierenDialog Methoden	10
2.3.5	ProfilBearbeitenDialog Methoden	10

1 Datenbankentwurf

1.1 Datenbankdiagramm

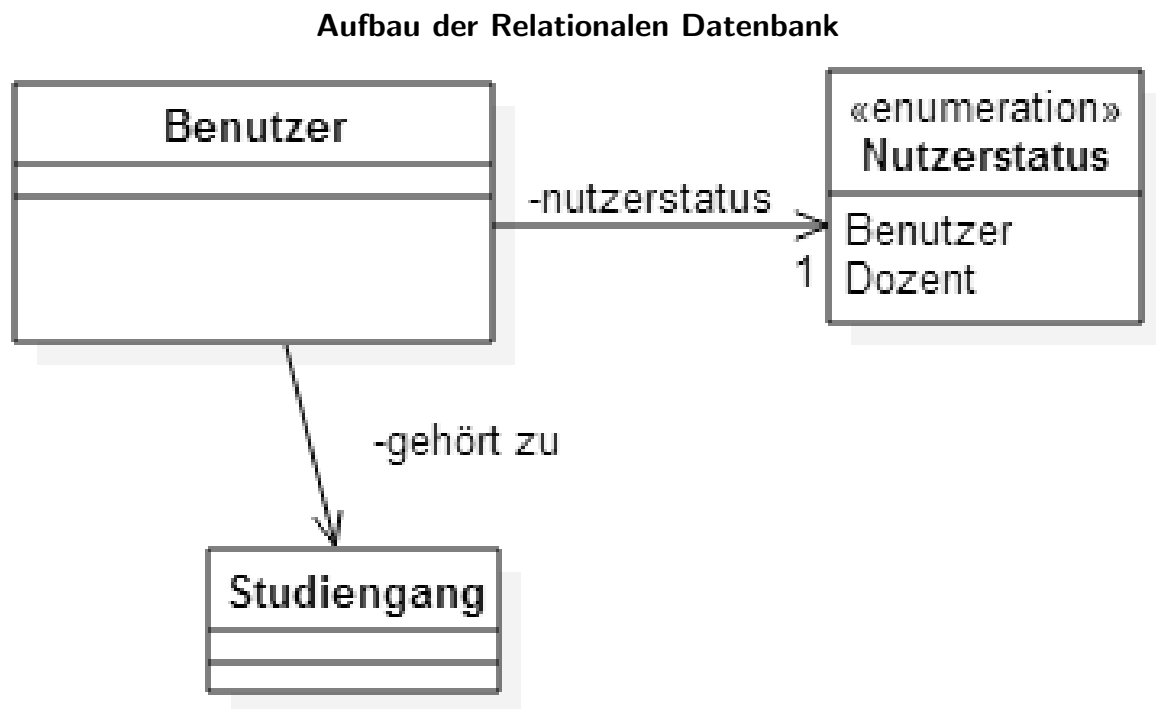


Abbildung 1.1: Datenbankdiagramm

1.2 Beschreibung der Tabellen

TABELLE	Benutzer
BESCHREIBUNG	Speichert alle Informationen, die zu einem Benutzer gehören.
VERWALTET	Benutzerdaten
SCHLÜSSEL	ID : Integer
FELD	Vorname : String
FELD	Nachname : String
FELD	Matrikelnummer : Integer
FELD	eMail : String
BESCHREIBUNG	Veranstaltungsbeschreibung
FELD	Studiengang : Int
BESCHREIBUNG	Fremdschlüssel, Benutzer ist bestimmtem Studiengang zugeordnet. Dozenten sind hier keinem Studiengang zugeordnet. Referenziert auf Tabelle Studiengang
FELD	Kennwport : String
BESCHREIBUNG	Speichert das verschlüsselte Passwort des Nutzers
FELD	Nutzerstatus : Enum
BESCHREIBUNG	Speichert ob Benutzer ein Student oder Dozent ist.
FELD	GruppeneinladungenErlauben : Bool
BESCHREIBUNG	Speichert ob Benutzer zu Gruppen eingeladen werden kann.
FELD	NotifyDiskussionen : Enum
BESCHREIBUNG	Speichert wie ein Benutzer über Diskussionen informiert wird.
TABELLE	Studiengang
BESCHREIBUNG	alle Studiengänge an der Universität Ulm
VERWALTET	Benutzer und deren eingetragenen Studiengang
SCHLÜSSEL	ID : Integer
FELD	Name : String
BESCHREIBUNG	Text der den Studiengang beschreibt.

2 System-Architektur

2.1 Kommunikationsdiagramme

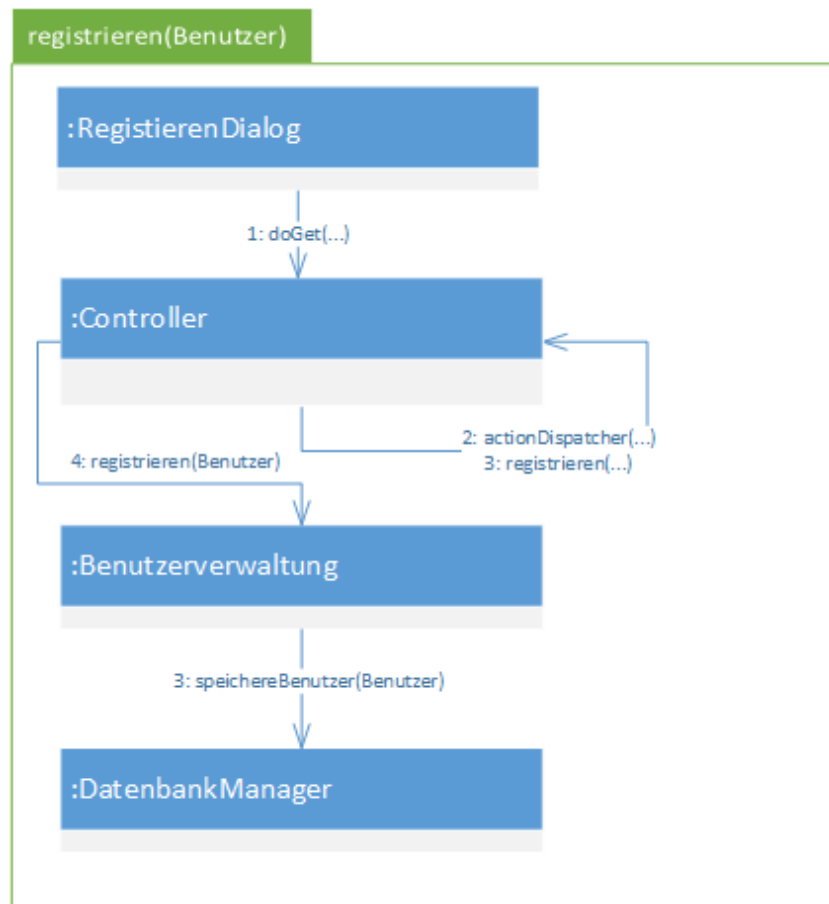


Abbildung 2.1: Registrierung

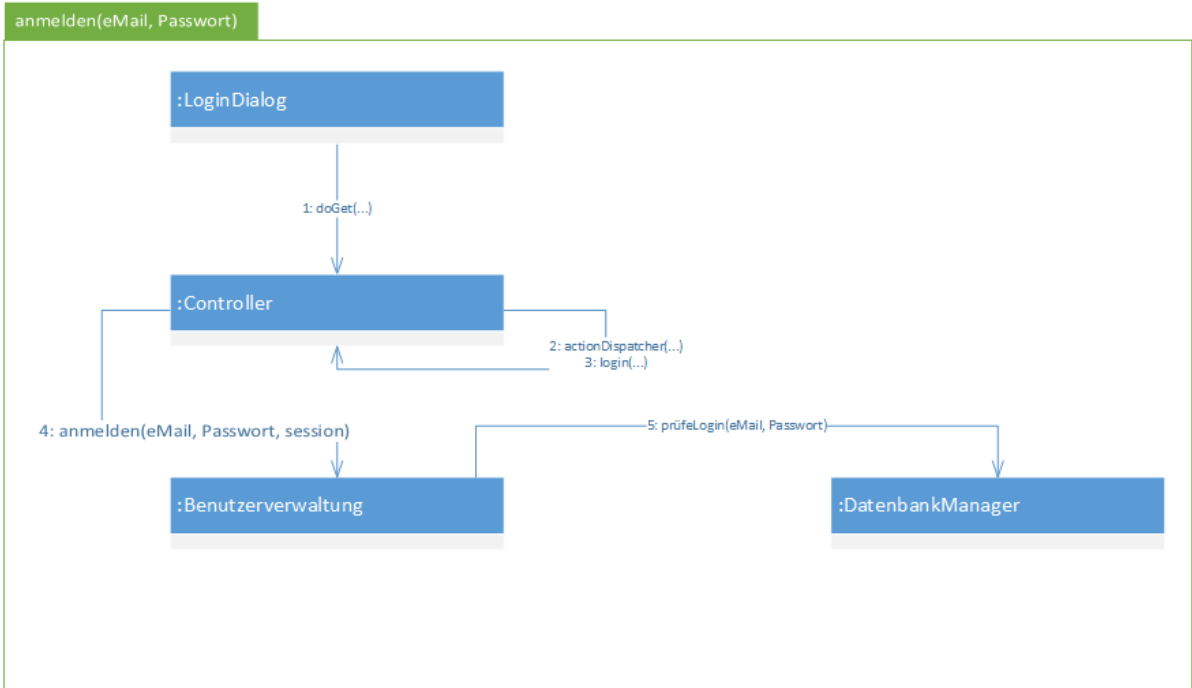


Abbildung 2.2: Am System anmelden

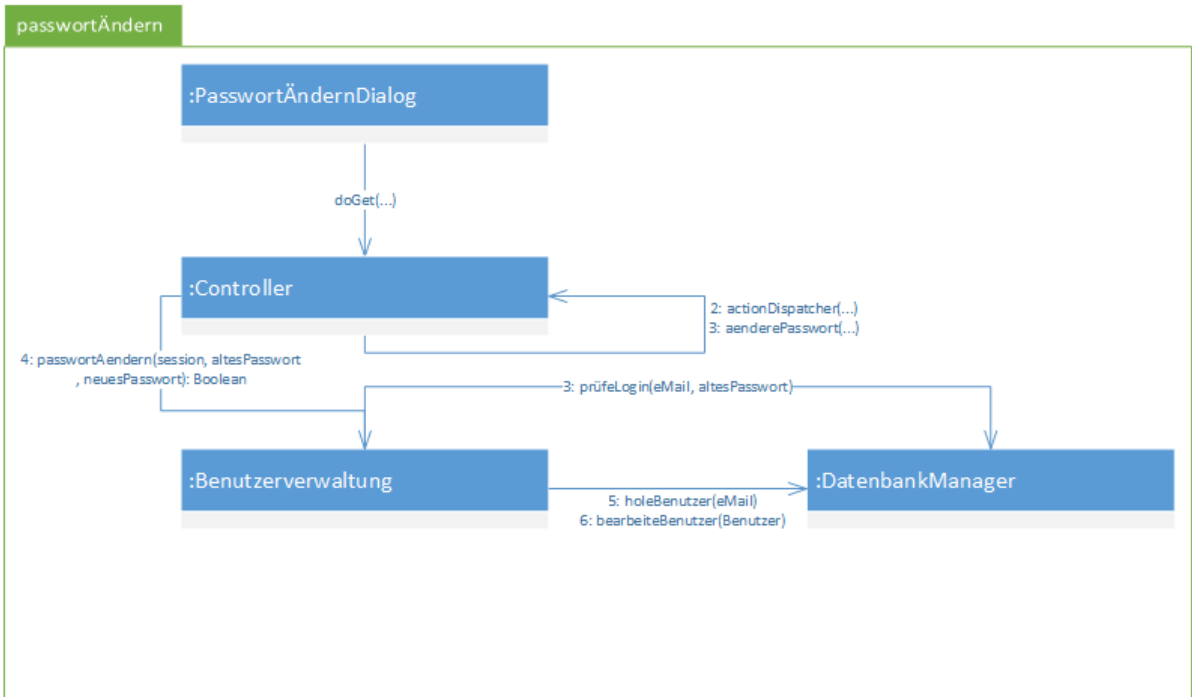


Abbildung 2.3: Passwortänderung

2.2 Klassendiagramm

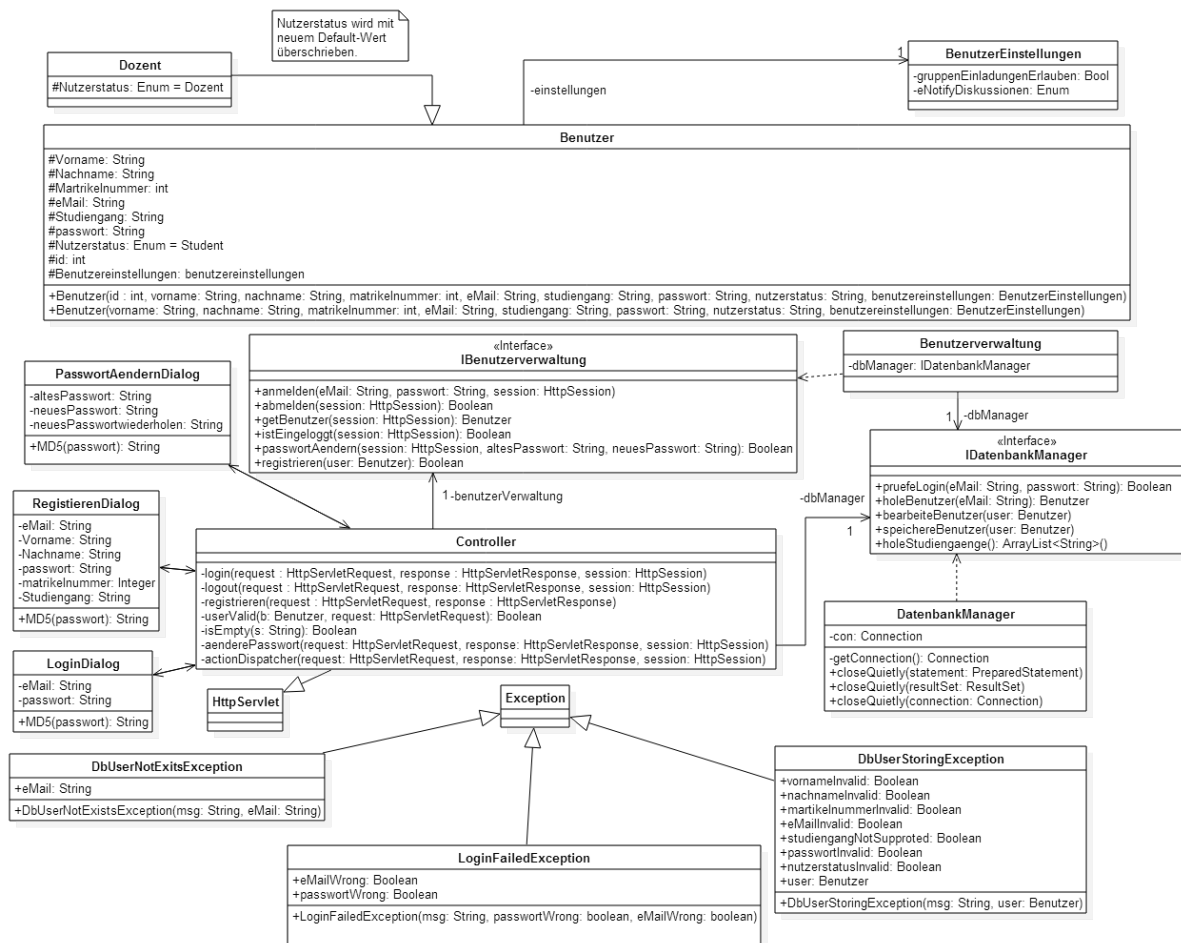


Abbildung 2.4: Klassendiagramm mit Realisierungsklassen

2.3 Methodenbeschreibung

2.3.1 Controller Methoden

Operation	doGet(HttpServletRequest request, HttpServletResponse response)
Beschreibung	Diese Methode nimmt einen ankommenden Request entgegen und gibt ihn an die Methode „actionDispatcher“ weiter.

Operation	actionDispatcher(HttpServletRequest request, HttpServletResponse response, HttpSession session)
Beschreibung	Diese Methode verwaltet die Weiterleitung zu den jeweils verantwortlichen Methoden für die Verarbeitung der Anfragen. Entsprechend zu der gewählten Action wird auf einen neuen Dialog weitergeleitet.

Operation	aenderePasswort(HttpServletRequest request, HttpServletResponse response, HttpSession session)
Beschreibung	Diese Methode übernimmt die Verarbeitung der Passwort-Änderungs-Anfragen. Es werden die Parameter aus dem Request ausgelesen. Wenn die empfangenen Parameter korrekt angegeben sind, wird die Anfrage an die Benutzerverwaltung weitergegeben. Je nachdem ob dies erfolgreich war, wird auf die Startseite weitergeleitet oder der Benutzer bekommt den gleichen Dialog nochmal angezeigt, wenn ein Fehler aufgetreten ist.

Operation	login(HttpServletRequest request, HttpServletResponse response, HttpSession session)
Beschreibung	Diese Methode übernimmt die Login-Anfragen. Sie liest die Parameter aus und leitet die Anfrage an die Benutzerverwaltung weiter. Es wird entweder auf die Startseite oder die Loginseite weitergeleitet.

Operation	logout(HttpServletRequest request, HttpServletResponse response, HttpSession session)
Beschreibung	Diese Methode übernimmt die Logout-Anfragen. Sie liest die Parameter aus und leitet die Anfrage an die Benutzerverwaltung weiter. Es wird auf die Loginseite weitergeleitet.

Operation	registrieren(HttpServletRequest request, HttpServletResponse response)
Beschreibung	Diese Methode übernimmt die Registrierungsanfragen. Sie liest die Parameter aus und leitet die Anfrage an die Benutzerverwaltung weiter. Es wird auf die Loginseite weitergeleitet.

Operation	userValid(Benutzer b, HttpServletRequest request): Boolean
Beschreibung	Diese Methode prüft, ob die gespeicherten Benutzerdaten gültig sind. Sie prüft ob alle Werte gesetzt sind und ob die eMail bspw. ein und einen Punkt enthält.

Operation	isEmpty(String s)
Beschreibung	Diese Methode prüft, ob der übergeben String null ist oder ob es ein Leerstring ist.

2.3.2 Benutzerverwaltung Methoden

Operation	getBenutzer(HttpSession: session): Benutzer
Beschreibung	Liefert das Benutzerobjekt zur entsprechenden Session zurück. Wenn die eMail-Adresse in der Session gespeichert ist, ist der Benutzer angemeldet. Dann wird das komplette Benutzerobjekt mit Hilfe des DB-Managers geladen und zurückgeliefert. Wenn ein Fehler auftritt wird null zurückgegeben.

Operation	anmelden(eMail:String, password:String, HttpSession: session)
Beschreibung	Meldet den Benutzer mit Passwort und eMail im System an. Die eMail wird in der aktuellen Session gespeichert um einen Verweis auf den Benutzer zu haben.

Operation	registrieren(user:Benutzer): Boolean
Beschreibung	Ruft beim Datenbankmanager „speichereBenutzer“auf. Wenn user null ist wird eine NullPointerException geworfen.

Operation	passwortAendern(HttpSession session, altesPasswort:String, neuesPasswort:String): Boolean
Beschreibung	Diese Methode ändert das Passwort für den aktuell angemeldeten Benutzer, indem es den aktuellen Benutzer mit „holeBenutzer“holt, das Passwort neu setzt und den Benutzer mit Hilfe des Datenbankmanagers mit „bearbeiteBenutzer“speichert.

Operation	abmelden(HttpSession session): Boolean
Beschreibung	Falls ein Benutzer in der übergebenen Session angemeldet ist, wird die eMail aus der Session entfernt und der Benutzer somit abgemeldet. Ist der Benutzer schon ausgeloggt wird einfach false zurückgegeben.

Operation	istEingeloggt(HttpSession session): Boolean
Beschreibung	Prüft ob die eMail in der aktuellen Session enthalten ist. Wenn dies zutrifft ist der Benutzer angemeldet.

2.3.3 Datenbankmanager Methoden

Operation	getConnection(): Connection
Beschreibung	Mit dieser Methode wird eine Verbindung zur Datenbank aufgebaut. Für alle Datenbankoperationen ist es eine Vorbedingung, dass eine Verbindung zur Datenbank existiert.

Operation	pruefeLogin(eMail:String, passwort:String): Boolean
Beschreibung	Es wird geprüft ob es in der Datenbank einen Benutzer mit der übergebenen eMail Adresse gibt. Als zweiter Schritt prüft die Methode, ob das Passwort zu diesem Benutzer passt. Wenn nicht wirft die Operation eine LoginFailedException. Die Exception bekommt als Parameter angegeben, ob die eMail oder das Passwort fehlerhaft ist. Sind die Benutzerdaten in Ordnung, dann gibt die Methode true zurück, ansonsten false.

Operation	holeBenutzer(eMail:String): Benutzer
Beschreibung	Vorbedingung: Benutzer in der Datenbank vorhanden. Die Operation liest aus der Datenbank den Benutzer mit der angegebenen eMail. Gibt es keinen passenden Benutzer in der Datenbank zu dieser eMail, dann wirft die Operation eine DbUserNotExistsException. Als Nachbedingung wird der Benutzer zurückgeliefert.

Operation	speichereBenutzer(user:Benutzer): Boolean
Beschreibung	Vorbedingung: Attribute des Benutzers sind alle korrekt. Beispielsweise dürfen Name und Vorname keine Leerstrings sein oder die eMail des Benutzers muss eindeutig sein. Außerdem muss der Studiengang des Benutzers in der Datenbanktabelle Studiengang vorhanden sein. Die Operation speichert den Benutzer in der Datenbank in der Tabelle Benutzer. Nachbedingung ist dass der Benutzer in der Datenbank gespeichert ist.

Operation	bearbeiteBenutzer(user:Benutzer)
Beschreibung	Vorbedingung: Benutzer ist in der Datenbank vorhanden, die Attribute des Benutzers sind alle korrekt und der Studiengang des Benutzers in der Datenbanktabelle Studiengang vorhanden sein. Die Operation ändert die Attribute des Benutzers in der Datenbank.

Operation	holeStudiengaenge(): List<String>
Beschreibung	Es gibt keine Vorbedingungen. Die Operation liest alle Studiengänge aus der Datenbank aus und gibt sie in einer Liste zurück.

2.3.4 Dialoge

Da sämtliche Dialoge nur HTML-Asichten im JSP-Format sind besitzen diese lediglich eine Javascript Hashfunktion um die eingegebenen Passwörter zu verschlüsseln.

Ansicht	Login-Dialog
Beschreibung	In dieser Ansicht hat der Benutzer die Möglichkeit sich mit seiner gewählten eMail-Adresse und dem dazugehörigem Passwort anzumelden, sollte das Kennwort oder die eMail-Adresse falsch oder nicht existent sein werden diese farblich markiert und eine Fehlermeldung ausgegeben.

Ansicht	Registrieren-Dialog
Beschreibung	In dieser Ansicht hat der Benutzer die Möglichkeit sich am System zu registrieren. Hierbei müssen alle gekennzeichneten Pflichtfelder ausgefüllt werden. Sollte der Button Registrierung abschließen betätigt wird der Benutzer wieder zum Login-Dialog geführt, sollten Fehler auftreten verbleibt der Benutzer auf dem Registrationsdialog.

Ansicht	Passwortänderungs-Dialog
Beschreibung	In dieser Ansicht kann der Benutzer sein gewähltes Passwort ändern. Sollten die Angaben fehlerhaft sein werden diese durch Fehlermeldungen oder Pop-up-Meldungen kenntlich gemacht. Ist die Änderung erfolgreich gelangt der Benutzer zum Login-Dialog, oder der Benutzer kann die Passwortänderung mit dem Button abbrechen.