

# Design-Beschreibung zu ELM

## 1 Allgemeines

ELM ist eine Software, die zum Online-Vorlesungs- und -Übungsbetrieb an einem Lehrstuhl/einer Fakultät, beitragen soll. Hierzu wird sie sowohl den Vorlesungsbetrieb, als auch den Übungsbetrieb verwalten.

Um mit dem Programm zu arbeiten wird ein Web-Browser genutzt, mit dem der Nutzer in die Lage versetzt wird, je nach Rolle und Rechten, im System bestimmte Aktionen durchzuführen. Die Realisierung des Programms erfolgt mittels Java Servlets. Die unterschiedlichen Daten, sowohl persönliche, als auch Vorlesungs- und Übungsspezifische, werden in einer Datenbank abgespeichert. Hierzu verwenden wir eine MySQL-Datenbank, es stellt jedoch kein Problem dar, eine andere Datenbank zu verwenden.

Diese Software dient im ersten Sinne der Übungsbetreuung eines Lehrstuhls, doch soll sie funktionabel genug sein um den Betrieb zu erweitern, z.B. auf eine Fakultät oder gar auf eine Universität oder eine ähnliche Hochschuleinrichtung.

Ein wichtiger Punkt des Programms ist aus Datenschutzgründen wichtige die Rechteverwaltung. Die meisten Rechte hat, verständlicherweise, der Admin und die wenigsten die StudentenInnen. Die möglichen Rechte sinken mit der Stufe des Nutzers.

## 2 Produktübersicht

Zum Produktumfang von ELM gehört die Verwaltung von Vorlesungen, von Übungen und der Online-Abgabe von Übungsaufgaben.

Dozenten, Übungsleiter und SHKs melden sich mit ihrem Benutzernamen am System an, die Studenten mit ihrer Matrikelnummer. Je nach Rolle und Rechten hat man nun die Wahl zwischen verschiedenen Services. Für Dozenten sind das z.B. das Anlegen von Vorlesungen und Übungen oder das Einrichten von Logins für Übungsleiter, für Übungsleiter z.B. das Einrichten von Seminaren oder das Einrichten von Logins für SHKs, für SHKs z.B. das Herunterladen von Lösungen oder das Eintragen von Punkten. Studenten haben z.B. die Möglichkeit, Vorlesungen und Übungen auszuwählen und sich zu Seminaren einzutragen oder Lösungen zu Übungen hochzuladen sowie sich die erreichten Punkte anzeigen zu lassen. Des weiteren gibt es die Rolle eines Administrators, der sämtliche Services der anderen Rollen nutzen kann.

## 3 Grundsätzliche Design-Entscheidungen

Die Software lässt sich von der Grundstruktur in 3 Kategorien unterteilen.

Als Grundlage dient das Basis-Paket, die Authentifizierung des Benutzers regelt und vor jedem Servlet-Aufruf überprüft, ob der Nutzer angemeldet ist und die entsprechenden Rechte zum Ausführen dieses Servlets besitzt. Dieses vorherige Überprüfen wird über Filter erreicht, welche bestimmte Session-Variablen überprüfen.

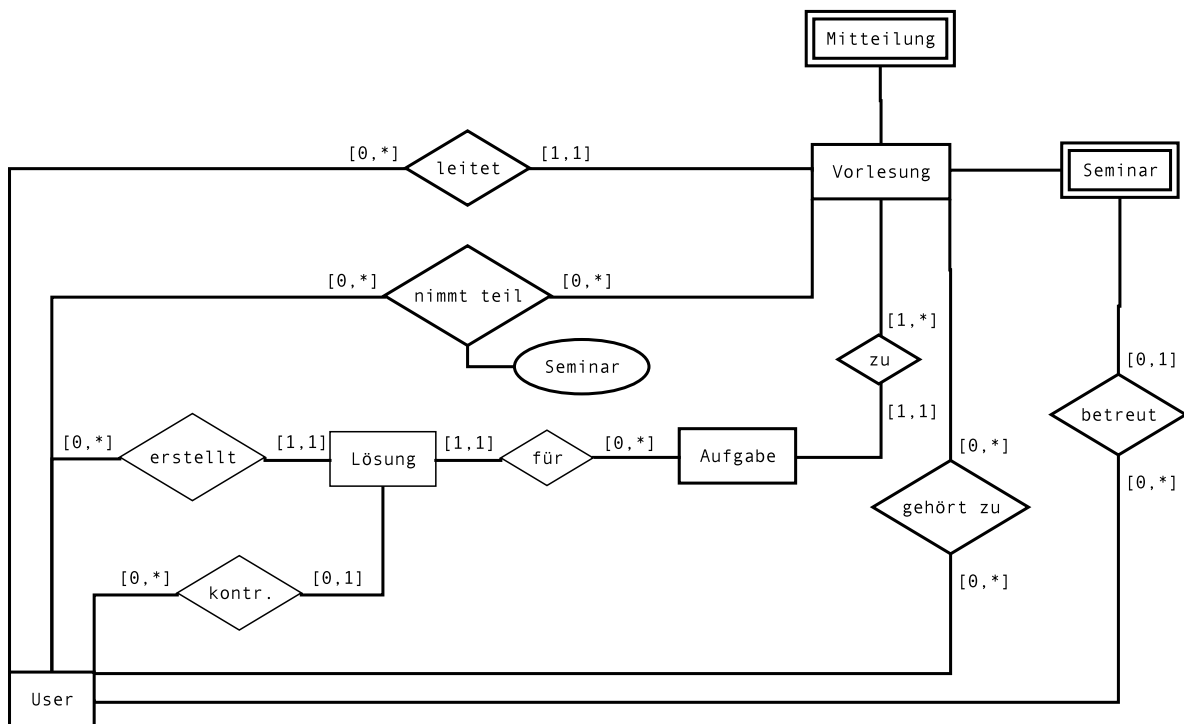
Die zweite Kategorie sind die Services, die für die Funktionalität der Software sorgen. Ein solcher Service besteht meist aus 2 Teilen. Der eine Teil stellt vorhandene Informationen bereit und bietet die Möglichkeit, neue Informationen in ein Formular einzugeben oder bestehende zu bearbeiten. Dieser Teil ist meist eine HTTP-GET-Methode. Der andere Teil ist dafür da,

Informationen, die im ersten Teil eingegeben wurden, zu prüfen und im System abzulegen. Dieser Teil ist in der Regel eine HTTP-POST-Methode.

Die dritte Kategorie bildet das Modell von ELM. Hierbei sind Objekte der realen Welt in Klassen abgebildet und können abgefragt und gesetzt werden. Die Daten bezieht die jeweilige Klasse aus der Datenbank und schreibt diese gegebenenfalls dahin zurück.

Welche Datenbank verwendet wird schreibt die Software nicht vor. Die Skripte zur Tabellen-Generierung sind für MySQL angepasst und müssten für andere Datenbanken ggf. leicht abgeändert werden. Der Zugriff auf die Datenbank erfolgt mit JDBC, für welches es für alle gängigen Datenbanken Treiber gibt.

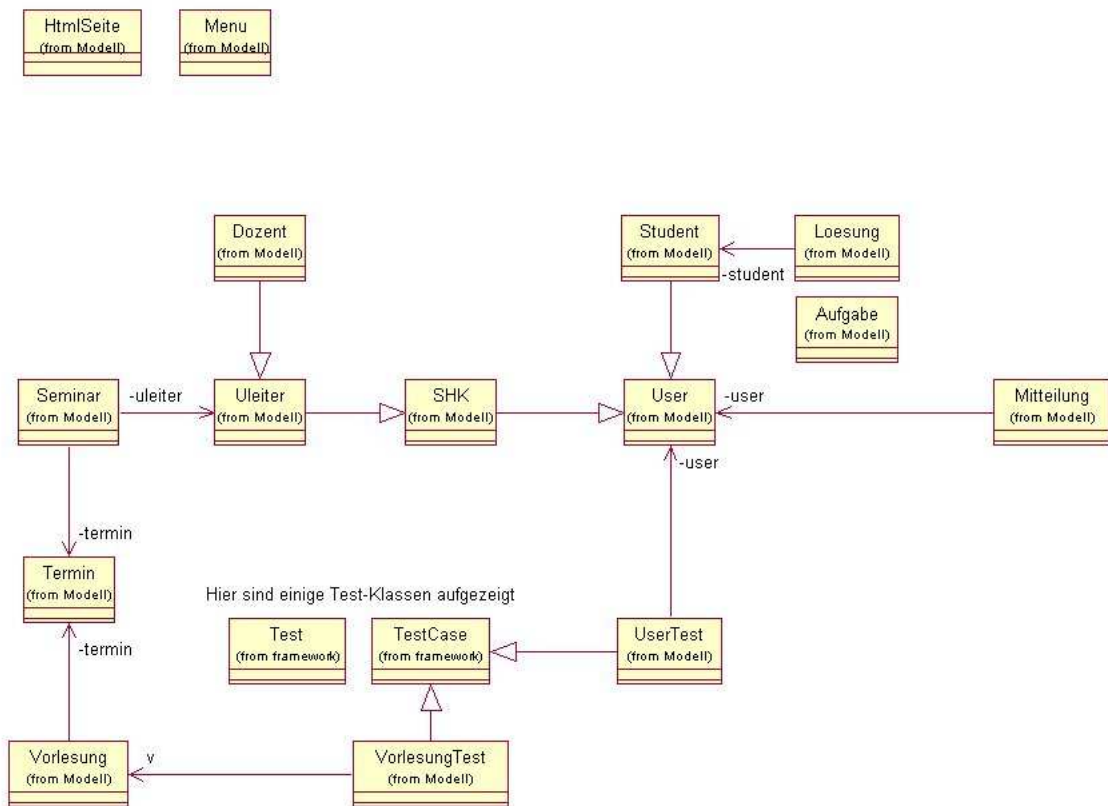
Die folgende Abbildung zeigt einen ER-Entwurf der Datenbank (größtenteils ohne Attribute).



## 4 Paket- und Klassenstruktur

Die Klassen von ELM sind in die folgenden Pakete unterteilt:

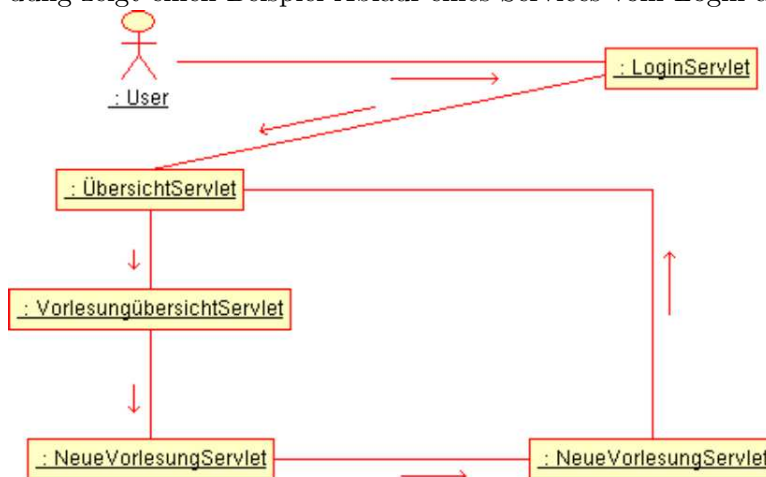
- ELM  
Dieses Paket ist die Wurzel von ELM. Innerhalb dieses Pakets befindet sich sämtliche weiteren Pakete und Klassen des Programms.
- ELM.Modell  
Die Klassen in diesem Paket bilden das Modell von ELM. Jede der Klassen realisiert ein Objekt aus der realen Welt. Die folgende Abbildung zeigt die Klassen dieses Pakets und ihre Verbindung untereinander.



Die Klassen holen sich die benötigten Informationen aus der Datenbank.

- User  
diese Klasse enthält login-Name und Passwort sowie Informationen über die Rechte des Users
- Dozent, Uleiter, SHK, Student  
enthält Zusatzinformationen der jeweiligen Rolle und Methoden zur Abfrage von z.B. verbunden Vorlesungen usw.

- Vorlesung  
enthält Methoden zum Abfragen und Setzen von Informationen zu einer Vorlesung, so z.B. den Dozent der Vorlesung und die Zeit der Veranstaltung
  - Seminar  
enthält Methoden zum Abfragen und Setzen von Informationen zu einem Seminar, so z.B. die zugehörige Übung, die Zeit der Veranstaltung oder die Liste der teilnehmenden Studenten
  - Aufgabe  
enthält Informationen zu einer Aufgabe, z.B. zu welcher Übung sie gehört oder wann der Abgabe-Termin ist
  - Loesung  
enthält Methoden, um eine Lösung hoch- und runterzuladen sowie Informationen, zu welcher Aufgabe die Lösung ist sowie das Abrufen von Informationen zu korrigierten Lösungen
  - Mitteilung  
enthält eine Mitteilung zu einer bestimmten Veranstaltung (Vorlesung oder Übung) oder erstellt eine
  - Termin  
Termin, zu dem eine Vorlesung oder ein Seminar stattfindet
  - HtmlSeite  
Klasse, die Methoden enthält, um die komplette HTML-Seite zu erstellen
  - Menu  
Diese Klasse generiert das dynamische Menü, das auf allen Seiten angezeigt wird.
- ELM.Service  
Jede Klasse innerhalb dieses Pakets oder eines seiner Unterpakete realisiert einen bestimmten Service, welcher in Form eines Servlets implementiert ist. Die folgende Abbildung zeigt einen Beispiel-Ablauf eines Services vom Login an.



Bei den Servlets auf der linken Seite handelt es sich um GET-Aufrufe, bei denen keine Daten vom Benutzer eingegeben wurden (nur Links bzw. Weiterleitungen). Bei den Servlets auf der rechten Seite handelt es sich um POST-Aufrufe, bei denen Daten vom Benutzer eingegeben wurden und die dort verarbeitet werden.

- Anmeldung, url: /anmeldung  
Service zum Neu-Anmelden eines Studenten
- Login, url: /login  
regelt den Login in ELM. Nach Authentifizierung liegt user in Session
- ServiceFilter, matcht: /elm/\*  
Dieser Filter regelt den Zugriff auf alle Services, die man nur im eingeloggen Zustand benutzen darf
- Logout, url: /elm/logout  
Loggt den User aus, zerstört die Session
- Optionen, url: /elm/option  
Service zum Ändern der eigenen Einstellungen, so email und Passwort
- MainServlet, url: /elm/main  
Übersichtsseite, die direkt nach dem Login angezeigt wird
- VorlesungsUebersicht, url: /elm/vorlesung  
Übersichtsseite zu einer bestimmten Vorlesung
- UebungsUebersicht, url: /elm/uebung  
Übersicht zur Übung zu einer bestimmten Vorlesung
- KlausurUebersicht, url: /elm/klausur  
Übersicht zur Klausur zu einer bestimmten Vorlesung
- Mitteilungen, url: /elm/mitteilungen  
Service zum Download einer Übungsaufgabe (Aufgabenstellung)

Die Services sind in 4 weitere Pakete unterteilt

- ELM.Service.Dozent

Dieses Paket enthält die Services für Dozenten.

- DozentFilter, matcht: \*.do  
Dieser Filter regelt den Zugriff auf alle Services, die man nur als Dozent erreichen darf
- NeueVorlesung, url: /elm/neuvorlesung.do  
Service zum Erstellen einer neuen Vorlesung

- ELM.Service.Uleiter

Dieses Paket enthält die Services für Übungsleiter

- UleiterFilter, matcht: \*.ul  
Dieser Filter regelt den Zugriff auf alle Services, die man nur als Übungsleiter erreichen darf
- NeuesSeminar, url: /elm/neuseminar.ul  
Service zum Erstellen eines neuen Seminars zu einer Vorlesung
- NeueUAufgabe, url: /elm/neuaufgabe.ul  
Service zum Erstellen einer neuen Übungsaufgabenserie zu einer Vorlesung
- NeuerMitarbeiter, url: /elm/neumitarbeiter.ul  
Service zum Erstellen eines neuen Mitarbeiters

- NeueMitteilung, url: /elm/neumitteilung.ul  
Service zum Erstellen einer neuen Mitteilung
- SeminarBearbeiten, url: /elm/seminarbearbeiten.ul  
Service zum Bearbeiten eines vorhandenen Seminars
- StudentenListe, url: /elm/studentenliste.ul  
Service zum Anzeigen der Studentenliste zu einer Vorlesung
- ELM.Service.SHK  
Dieses Paket enthält die Services für SHKs
  - SHKFilter, matcht: \*.sh  
Dieser Filter regelt den Zugriff auf alle Services, die man nur als SHK erreichen darf
  - LoesungDownload, url: /elm/downloadloesung.sh  
Service zum Download mehrerer Lösungen von Studenten zu einer Aufgabenserie in einem Zip-File
  - PunkteUpload, url: /elm/uploadpunkte.sh  
Service zum Eintragen der Punkte von Lösungen
- ELM.Service.Student  
Dieses Paket enthält die Services für Studenten
  - StudentFilter, matcht: \*.st  
Dieser Filter regelt den Zugriff auf alle Services, die man nur als Student erreichen darf
  - VorlesungAuswahl, url: /elm/vorlesungsauswahl.st  
Service zur Auswahl der eigenen Vorlesungen
  - SeminarAuswahl, url: /elm/seminarauswahl.st  
Service zur Auswahl eines Seminars zu einer Vorlesung
  - NeueLoesung, url: /elm/neuloesung.st  
Service zum Hochladen einer neuen Lösung zu einer Aufgabenserie

Dabei können Dozenten auch Services von Übungsleitern und SHKs verwenden und Übungsleiter Services von SHKs.