

Designbeschreibung zu WebAssign

1. Allgemeines

Das Softwareprodukt WebAssign wurde von der Fern – Universität Hagen entwickelt, um angesichts des Trends in die virtuelle Lehre an Hochschulen und dem Bedarf an qualitativ hochwertiger verteilter Weiterbildung in der Industrie gerecht zu werden. WebAssign ist Opensource-Produkt, es ist in Java implementiert, läuft unter Linux und Solaris und benutzt Opensource-Software (Apache, Tomcat, MySQL) sowie die Softwarestandards JDBC und CORBA. Als Datenbank können auch ORACLE und Informix verwendet werden.

2. Produktübersicht

2.1 Produktbeschreibung

WebAssign realisiert einen umfassenden universitären Übungsbetrieb. Im WebAssign-System greifen alle beteiligten Nutzer über eine Web-Schnittstelle auf ein zentral installiertes Anwendungssystem zu. Dort werden alle anfallenden Daten und Dokumente in einer Datenbank verwaltet. WebAssign unterstützt alle Nutzer im Prozess der Erstellung und Bearbeitung von Einsendeaufgaben.

Drei verschiedene Gruppen von Nutzern gilt es zu unterscheiden:

1. Student: Er greift auf Aufgabenseiten zu, sendet Lösungen dazu ein und kann seine Korrekturen und erreichte Punkte einsehen, sich für Klausuren anmelden.
2. Betreuer: Er entwickelt Aufgaben und das dazugehörige Design des Aufgabenmaterials, schreibt Studenten und Korrektoren ein und kontrolliert die Arbeit der Kontrolleure.
3. Korrektor: Er überprüft die Aufgabenlösungen der Studenten und versieht diese Lösungen mit Anmerkungen und trägt die erreichte Punktzahl ein.

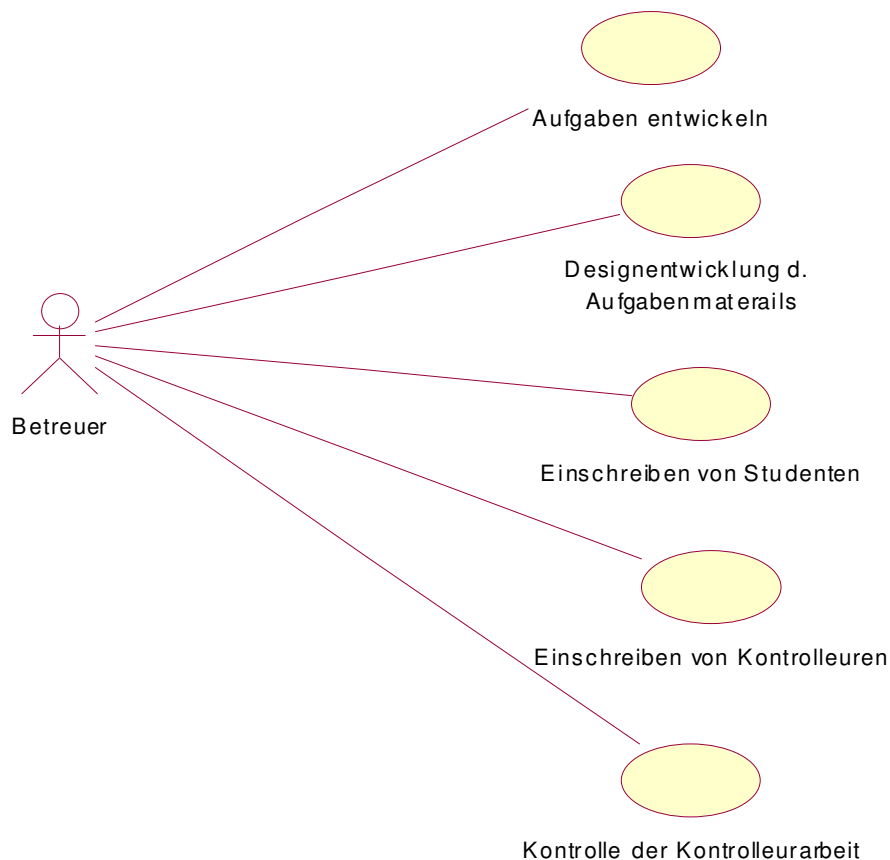
Darüber hinaus soll zwischen angemeldeten Nutzern der Versand von Nachrichten in Form eines Online-Chat möglich sein.

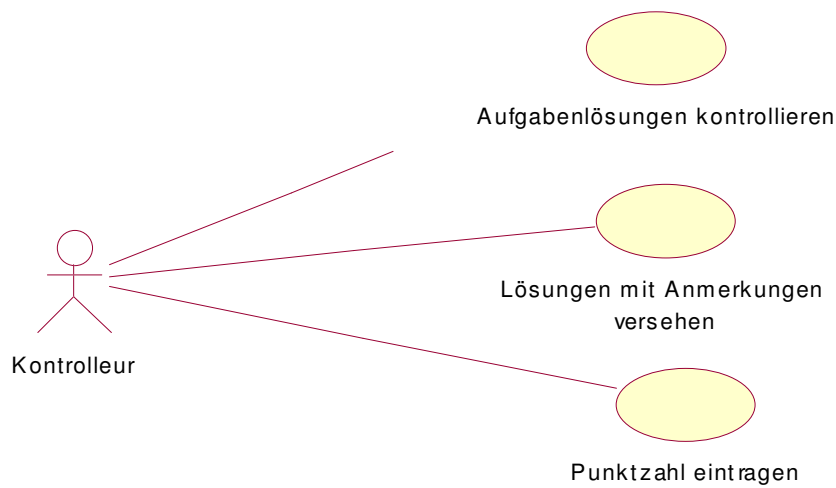
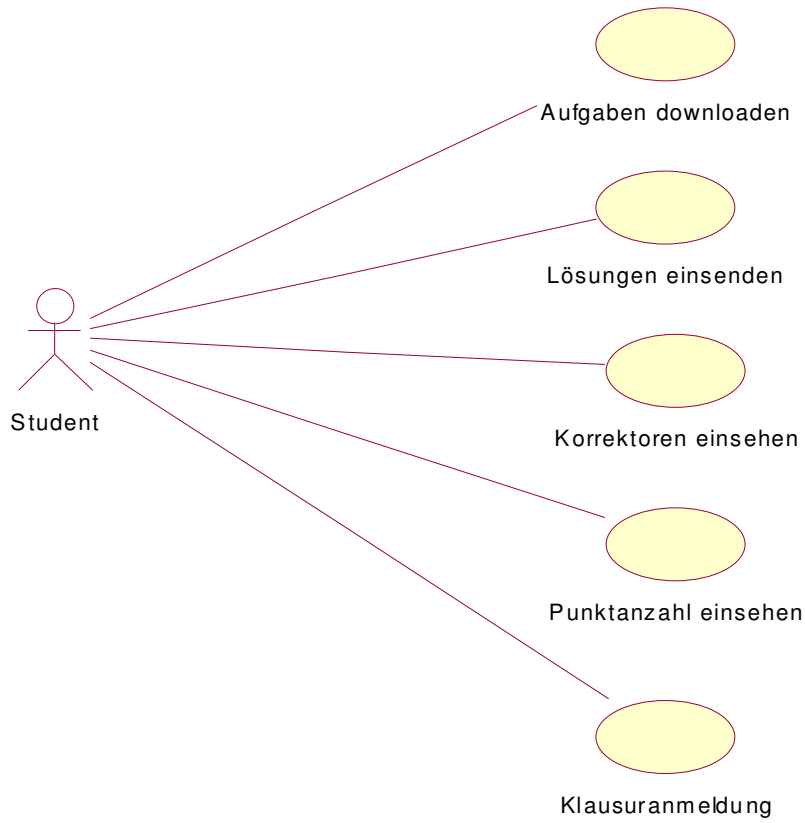
Die Kursstartseite ist die zentrale Einstiegsseite für alle Beteiligten; eine Prüfung der Zugangsberechtigung findet erst bei den von hier aus erreichbaren Seiten statt. Die Kursstartseite kann den Erfordernissen des jeweiligen Kurses angepasst werden. Auf der Studentenstartseite sieht man eine nach Aufgabenheften geordnete Übersicht aller schon bearbeiteten beziehungsweise noch zu bearbeitenden Aufgaben. Für Aufgabenhefte, deren Bearbeitungszeitraum noch nicht begonnen hat, sind die einzelnen Aufgaben noch nicht zugänglich. Über die jeweiligen Aufgaben - Links gelangt man zum eigentlichen Aufgabentext. Auch die Studentenseite kann bei Bedarf den Erfordernissen des Kurses angepasst werden.

Auf der Aufgabenseite befindet sich der eigentliche Aufgabentext. Durch das Eintragen von Lösungen in die dafür vorgesehenen Felder wird die Aufgabe bearbeitet und durch Betätigen des Einsendeknopfes zum WebAssign - Server geschickt.

Endgültig abgeschlossen ist die Aufgabe jedoch erst, wenn das gesamte Aufgabenheft geschlossen oder das Bearbeitungsende zeitlich überschritten ist. Bis zu diesem Zeitpunkt besteht die Möglichkeit, die Aufgabe beliebig oft einzusenden. Zur Überprüfung seiner persönlichen Ergebnisse erhält der Studierende eine Auflistung seiner Korrekturen mit dem jeweiligen Status und gegebenenfalls den erreichten Punkten sowie Links zu den einzelnen Musterlösungen. Diese Seite ist personalisiert und nur ihm zugänglich. Von der Betreuerstartseite aus können alle administrativen Funktionen erreicht werden. Beispielsweise findet sich hier eine Aufgabenübersicht, die zu Konfigurationsseiten für die einzelnen Aufgaben führt. Eine Einsendungs- und Korrekturübersicht gibt einen nach Aufgabenheften und Korrektoren geordneten Überblick, wie viele Aufgaben welchen Status haben. Die Betreuerstartseite und deren Folgeseiten sind passwortgeschützt nur für Betreuer zugänglich.

Auf der Startseite der Korrektoren sieht ein Korrektor die ihm zugeordneten Aufgaben im Überblick. Nach Aufgabenheften geordnet wird angezeigt, wie viele Aufgaben welchen Status haben. Auch diese Seite ist passwortgeschützt nur Korrektoren zugänglich.





2.2 Produktanforderungen

Die Anwendung ist ein über das Netzwerk verteiltes System. Dadurch greifen gleichzeitig mehrere Endbenutzer auf die Anwendung zu, das heißt es ist ein Programm mit der Fähigkeit zur Mehrbenutzerverwaltung.

3. Grundsätzliche Designbeschreibung

3.1 Verteilungsarchitektur

Das Produkt wurde im Rahmen einer Web-Architektur modelliert, das heißt Client und Server kommunizieren über das http -Protokoll auf Grundlage einer temporär hergestellten Verbindung miteinander. Auf der Client-Seite wird somit nur ein Webbrowser benötigt. Der restliche Teil der Anwendung läuft komplett serverseitig ab.

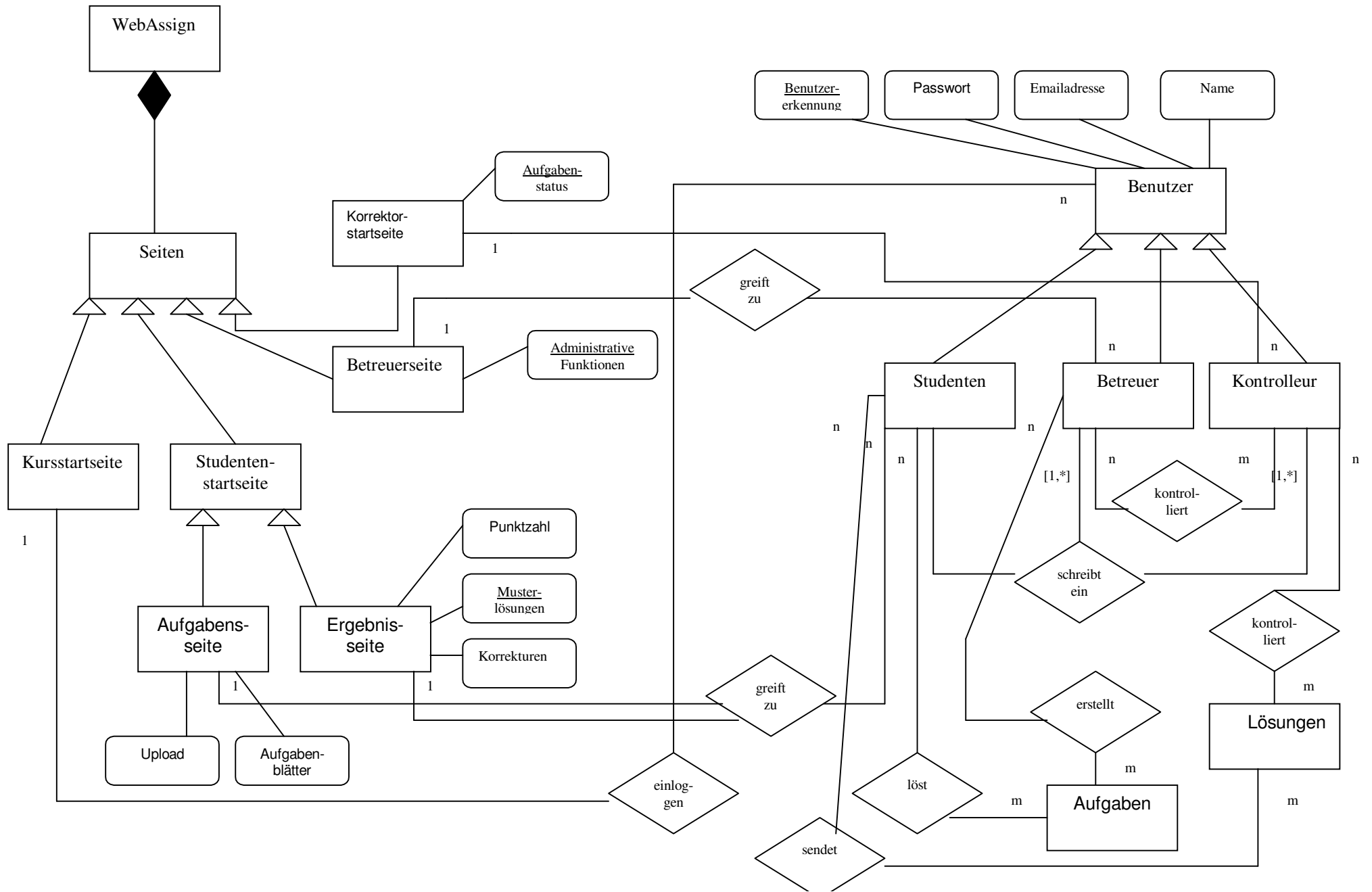
3.2 Benutzeroberfläche

Die Benutzungsoberfläche ist als GUI (graphical user interface) konzipiert. Hierbei kann der Benutzer der Anwendung diese steuern, indem er Eingaben in die graphische Benutzeroberfläche mittels Tastatur oder Maus macht. Diese empfängt das GUI - System, wertet sie aus und leitet sie an die eigentliche Anwendung weiter. Diese Anwendung reagiert dann auf diese Eingaben in der implementierten Vorgehensweise.

Dabei findet sowohl die Kommunikation zwischen GUI und Endnutzer als auch zwischen GUI und Anwendung durch das Versenden von HTML-Code statt.

3.3 statische Beschreibung

ER Diagramm: Nach dem wir die Daten bezüglich der Anforderungen, von WebAssign, analysiert haben, haben wir folgendes ER -Diagramm entworfen.



Verantwortlicher: Peter Scholz

Bearbeitet: Denny Sindermann, Julian Hesselbach, Peter Scholz

Gruppe: Ueb 13

3.Aufgabenblatt

25.05.03

4. Paket u. Klassenstruktur

Als mdl – Datei (wa_long-all.mdl) mit beigefügt.