

Übungen im Internet



Handbuch für Autoren

Stand: April 2004

FernUniversität Hagen

URZ

Copyright © 2001 CampusSource, <http://www.campussource.de>

1	Technische Voraussetzungen	3
2	Zugang zu WebAssign	4
3	Aufgabenhefterstellung	6
4	Aufgabenerstellung	8
4.1	Verschiedene Aufgabentypen	8
4.2	Aufgabenstruktur	9
4.3	Aufgabenerstellung allgemein	9
4.4	Der Aufgabenerstellungs-Assistent für automatisch auswertbare Aufgaben	14
4.4.1	<i>Multiple Choice mit Einfach- und Mehrfachauswahl</i>	18
4.4.2	<i>Zuordnungsaufgaben</i>	22
4.4.3	<i>Zahl-Aufgaben</i>	24
4.4.4	<i>Begriffsabfrage</i>	25
4.4.5	<i>Sichern und Ändern der Aufgaben</i>	27
4.5	Frei-Text-Aufgaben	27
4.5.1	<i>Variablen in Aufgabentexten</i>	27
4.5.2	<i>Aufgabe analysieren und Teilaufgaben erzeugen</i>	29
4.5.3	<i>Anlegen weiterer Eingabefelder</i>	30
4.5.4	<i>Anlegen weiterer Einsendebutton</i>	33
4.6	Bewertungs- und Selbsttestmodule	34
4.7	Automatisch auswertbare Aufgaben ohne den Aufgabenerstellungs-Assistent	35
4.8	Programmieraufgaben mit Pre-Testing	35
4.9	Fortgeschrittene Gestaltung des Aufgabentexts.....	36
4.9.1	<i>Verwendung von Grafiken und Applets in Aufgaben</i>	36
4.9.2	<i>Verwendung von Grafiken in Quittungsseiten</i>	37
4.10	Aufgaben, zu denen Studierenden mit beliebigen Editoren die Lösung erstellen	38
4.10.1	<i>Einbindung der studentischen Lösung zur erneuten Bearbeitung</i>	39
4.10.2	<i>Korrekturschablonen</i>	40
4.10.3	<i>Quittungen</i>	40
4.11	Aufgaben, die von Autoren mit beliebigen Editoren erstellt werden	40
4.12	Konsistenzerhaltung nach Änderungen einer Aufgabe.....	41
4.13	Publizieren einer Aufgabe.....	41
4.14	Korrekturseite.....	42
4.15	Quittungen.....	44
4.16	Musterlösung.....	45
5	Aufgaben überprüfen und testen.....	45

WebAssign

Handbuch für Autoren

Dieses Handbuch erklärt Ihnen, wie Sie in WebAssign:

- Zugang als Autor erhalten,
- Aufgaben verwalten,
- den Aufgabentyp festlegen,
- Aufgabentexte eingeben,
- den Korrekturmodus festlegen,
- Quittungen, Korrekturschablonen und Musterlösungen anlegen,
- das Kursressourcen-Verzeichnis verwalten,
- Bilder, Graphiken und Applets in Aufgabentexte und Musterlösungen einbinden sowie
- die Formatierung der erstellten Aufgabentexte überprüfen können.

Generelle Annahme:

Im Handbuch für Autoren gehen wir davon aus, dass die Kursbetreuer die Aufgabenformate vorgegeben haben, um z.B. bei mehreren Autoren für ein einheitliches Erscheinungsbild der Aufgaben zu sorgen. Wenn Sie über die hier beschriebenen Funktionalitäten hinausgehende Kenntnisse¹ benötigen, lesen Sie bitte im *Handbuch für Betreuer* nach.

Wegweiser für Autoren

Als Autor müssen Sie wissen, wie Sie Zugang zu WebAssign erhalten und Aufgaben anlegen, lesen Sie hierzu die Abschnitte 2 und 3, 4.2 und 4.3.

Wenn Sie automatisch auswertbare bzw. Lotse-Aufgaben erstellen, lesen Sie den Abschnitt 4.4.

Wenn Sie offene Aufgaben, d.h. Aufgaben, bei denen die Studierenden Texte einsenden, erstellen wollen, lesen Sie die Abschnitte 4.5.

Wenn Sie Aufgaben mit Pre-Testing erstellen, lesen Sie im Abschnitt 4.8.

In allen Fällen müssen Sie die erstellten Aufgaben aus Studierender- und gegebenenfalls aus Korrektorsicht testen. Lesen Sie hierzu den Abschnitt 5 und lesen Sie die *Handbücher für Studierende und Korrektoren*. Wenn Sie den Korrekturvorgang prüfen, lesen Sie im *Handbuch für Betreuer* wie Betreuer Korrekturen verwalten.

1 Technische Voraussetzungen

Um WebAssign nutzen zu können, brauchen Sie einen Web-Browser. Erprobt sind die Versionen 3 und 4 des Netscape Browsers.

1. Alle Funktionalitäten, die fortgeschrittene Kenntnisse in HTML erfordern, sind im Handbuch für Betreuer beschrieben.

2 Zugang zu WebAssign

Ihre Kursbetreuer teilen Ihnen die www-Adresse für die Kurse, zu denen Sie Aufgaben erstellen, mit. Für Angehörige der FernUniversität Hagen gilt die Adresse¹

<https://vu.fernuni-hagen.de/wb/webassign/veranstalter/KursStartSeite/Kursnummer/Semester>.

Desweiteren benötigen Sie eine Benutzerkennung und ein Passwort. Diese bekommen Sie ebenfalls von Ihren Kursbetreuern zugeteilt.

Als Autor benutzen Sie den gleichen Zugang wie die Betreuer eines Kurses (siehe Abb. 1). Dieses hat historische Gründe: an der FernUniversität Hagen haben die Betreuer immer auch die Autorenrolle, d.h. die Aufgabenverwaltung übernommen. Die Aufgabenverwaltung finden Sie daher unter dem *Zugang für Betreuer* (siehe Abb. 2). (Alle anderen Funktionalitäten finden Sie im *Handbuch für Betreuer* erklärt.)

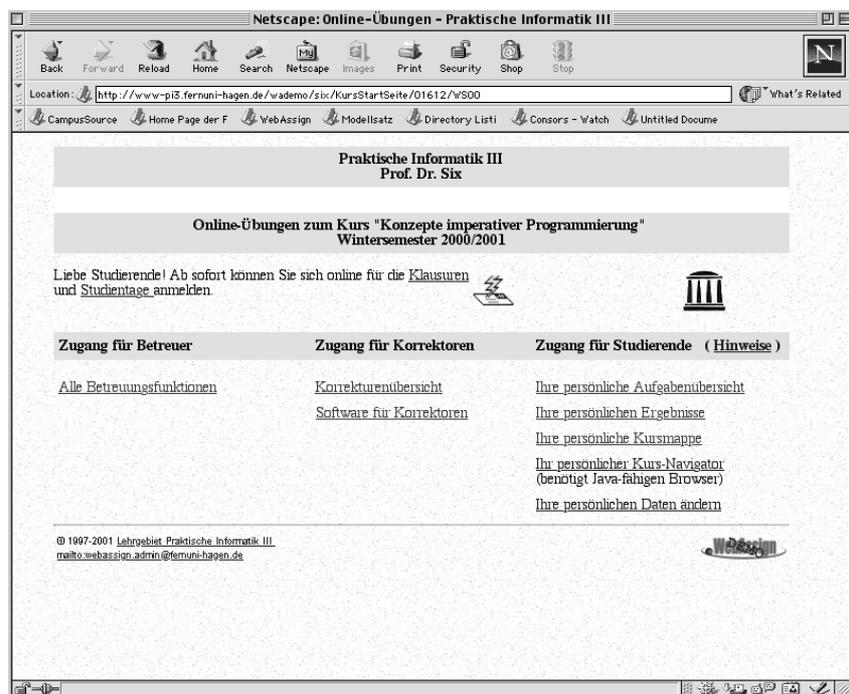


Abb. 1: Die Kursstartseite: Mit Ihrer Benutzerkennung und Ihrem Passwort können Sie den Zugang für Betreuer auswählen

Die Betreuerstartseite (siehe Abb. 2) ist die zentrale Anlaufstelle für Autoren und Kursbetreuer. Hier können Sie alle administrativen Aufgaben² erledigen.

1. Der Zugang zu den Übungen über die VU ist nur für Studierende möglich; d.h. als Autor gilt Ihre Benutzerkennung hier nicht.
2. In diesem Handbuch gehen wir davon aus, dass Sie als Autor online arbeiten. Wenn Sie offline arbeiten möchten, lesen Sie bitte im Betreuerhandbuch die entsprechenden Kapitel über das Herunterladen und Heraufladen von Kursressourcen nach. Um offline Aufgaben zu erstellen, sollten Sie HTML sicher beherrschen.

In der Rubrik *Aufgabenverwaltung* sind die Funktionen für Autoren aufgelistet.

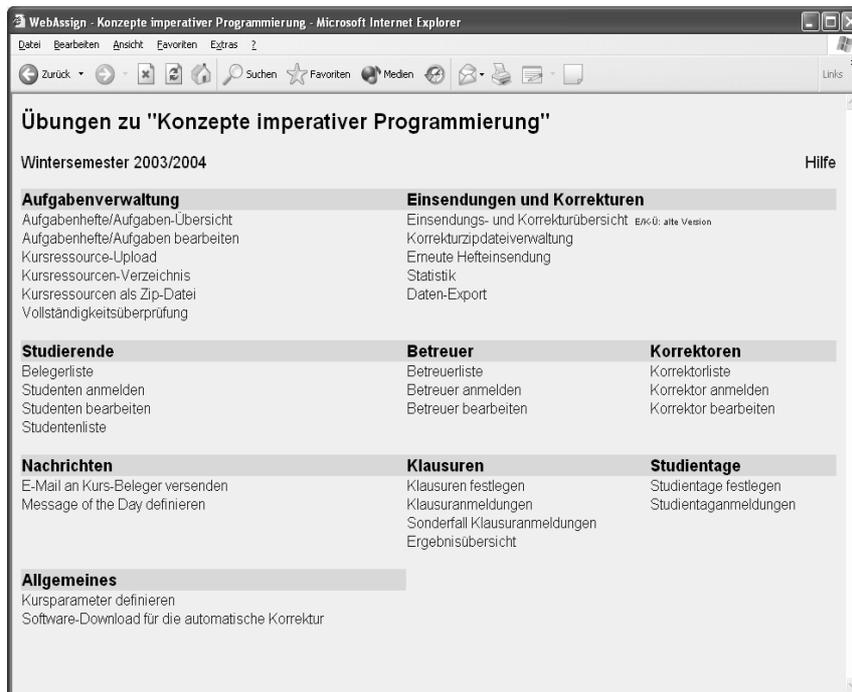


Abb. 2: *Gemeinsame Übersicht über alle Funktionen für Autoren und Betreuer.*

In den nächsten Kapiteln werden die einzelnen Funktionalitäten näher erläutert.

3 Aufgabenhefterstellung

FernUniversität Hagen
Praktische Informatik III
Prof. Dr. Six

Übungen zu "Konzepte imperativer Programmierung"

Wintersemester 2003/2004

[Kursressourcen-Verzeichnis](#) [Kurs-Startseite](#) [Hilfe zur Kursbetreuung](#)
[Kursressourcen als Zip-Datei](#) [Betreuer-Startseite](#) [Hilfe zur Aufgabenerstellung](#)
[Kursressource-Upload](#) [Studenten-Startseite](#)

Aufgabenhefte	Beginn	Ende	Aufgaben	Korrekturfreigabe	Musterlösung
Aufgabenheft 1	01.09.2003, 12:00	12.01.2004, 12:00	Aufgabe 1 (UE) Aufgabe 2 (UB) Aufgabe 3 (UB) Aufgabe 4 (UE) Aufgabe 5 (UE)	nach Ende	nach Ende
Aufgabenheft 2	22.09.2003, 12:00	21.10.2003, 12:00	Aufgabe 1 (UE) Aufgabe 2 (HB) Aufgabe 3 (UB) Aufgabe 4 (AB) Aufgabe 5 (AB) Aufgabe 6 (AB) Aufgabe 7 (AB)	nach Ende	nach Ende
Aufgabenheft 3	06.10.2003, 12:00	03.11.2003, 12:00	Aufgabe 1 (UB) Aufgabe 2 (UB) Aufgabe 3 (UB) Aufgabe 4 (HB) Aufgabe 5 (AB) Aufgabe 6 (AB) Aufgabe 7 (AB) Aufgabe 8 (AB) Aufgabe 9 (AB)	nach Ende	nach Ende
Aufgabenheft 4	20.10.2003, 12:00	19.11.2003, 12:00	Aufgabe 1 (UE) Aufgabe 2 (HB) Aufgabe 3 (UB) Aufgabe 4 (AB) Aufgabe 5 (AB)	nach Ende	nach Ende

Abb. 3: Übersicht über alle Aufgabenhefte einschließlich der Aufgaben.

Wählen Sie den Link *Aufgabenhefte/Aufgaben-Übersicht* aus, so finden Sie die Aufgaben nach Aufgabenheften geordnet (siehe Abb. 3). In der ersten Spalte finden Sie die fest vorgegebene Nummer des Aufgabenhefts als Link. Der Link führt zur Aufgabenheftverwaltung. Dort können Sie die Eigenschaften des gewählten Aufgabenhefts ändern. In der zweiten Spalte steht der Bearbeitungsbeginn. Zu diesem Zeitpunkt werden die Aufgaben für die Studierenden zur Bearbeitung freigegeben. Vor diesem Termin kann ein Studierender sich die Aufgabenstellungen nicht anschauen. Nach dem Bearbeitungsende, welcher in der dritten Spalte aufgeführt ist, können die Studierenden keine weiteren Lösungen mehr einsenden. In der vierten Spalte stehen alle Aufgaben, die dem Aufgabenheft zugeordnet sind. Für jede Aufgabe gibt es einen Link über den Sie zum Aufgabentext und zur Konfiguration der Aufgabe kommen. In der fünften Spalte wird angegeben, ob erfolgte Korrekturen und Bewertungen sowie die Musterlösungen bereits vor dem allgemeinen Bearbeitungsende für die Studierenden frei gegeben werden. Dies ist besonders für automatische und damit schnelle Korrekturen zu beachten. Bis auf die Freigabe der Musterlösung werden diese Informationen über die Aufgabenheftverwaltung definiert. Die Freigabepunkt der Musterlösung wird über die Betreuerfunktionalität *Kursparameter definieren* festgelegt. (s. Handbuch für Betreuer)

Nach Anklicken einer Aufgabenheftnummer gelangen Sie zur Aufgabenheftverwaltung. Alternativ können Sie auf der Betreuerstartseite den Link *Aufgabenhefte/Aufgaben bearbeiten* wählen. WebAssign merkt sich das zuletzt bearbeitete Aufgabenheft und führt Sie wieder zu dessen Aufgabenheftverwaltungsseite. Abb. 4 zeigt solch eine typische Seite:

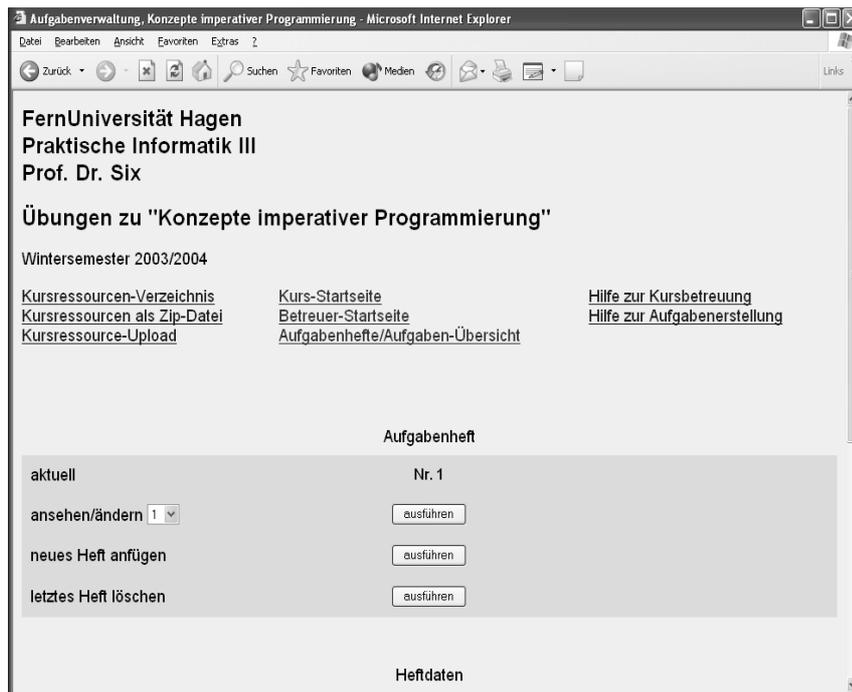


Abb. 4: Aufgabenheftverwaltung

Im oberen Teil der Seite finden Sie Informationen und Aktionen, die alle Aufgabenhefte betreffen. In der Zeile *aktuell* sehen Sie die Nummer des Hefts, welches aktuell bearbeitet wird. Mit *ansetzen/ändern* können Sie zwischen den Aufgabenheften wechseln. Mit *neues Heft anfügen* erzeugen Sie ein neues Aufgabenheft. Sie können zur Zeit Aufgabenhefte nur am Ende anfügen, d.h. Sie können kein neues Heft zwischen zwei bestehenden Heften einfügen. Wenn Sie ein Aufgabenheft zwischen bereits bestehende einfügen möchten, arbeiten Sie offline auf den Kursressourcen (siehe *Handbuch für Betreuer*) oder wenden Sie sich an Ihre Betreuer. Die Aufgabenhefte werden automatisch durchnummeriert. In neu erzeugten Aufgabenheften sind zunächst keine Aufgaben angelegt und kein Bearbeitungszeitraum festgesetzt.

Mit *letztes Heft löschen* wird das letzte Aufgabenheft gelöscht. Wenn Sie ein beliebiges Aufgabenheft löschen wollen, arbeiten Sie offline auf den Kursressourcen (siehe *Handbuch für Betreuer*) oder wenden Sie sich an Ihre Kursbetreuer.

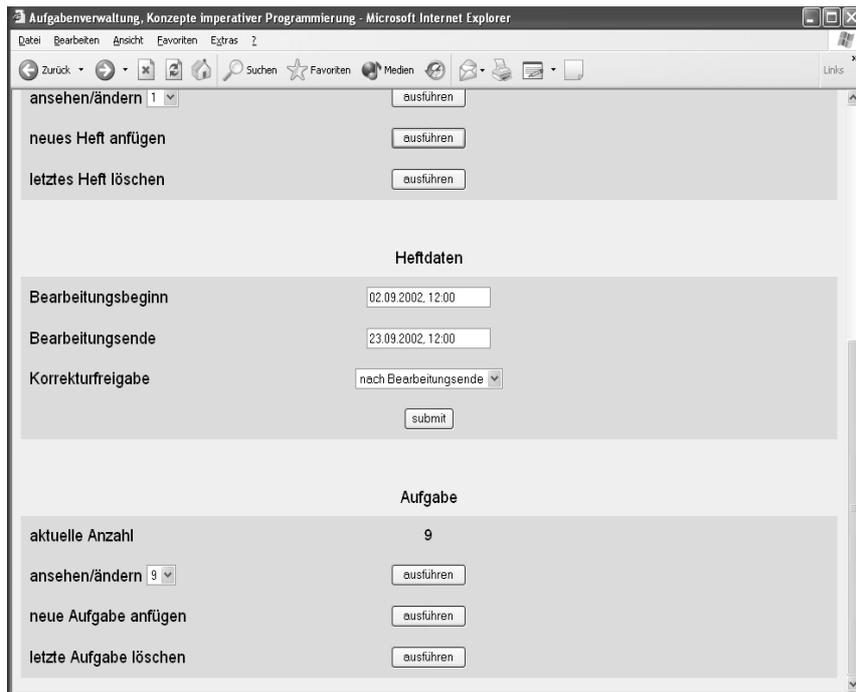


Abb. 5: Aufgabenheftverwaltung, Fortsetzung

Im Bereich *Heftdaten* der Aufgabenheftverwaltung setzen Sie den Bearbeitungsbeginn, das Bearbeitungsende des Hefts und legen fest, zu welchem Zeitpunkt die Freigabe der Korrektur erfolgen soll. Durch Drücken des submit-Buttons speichern Sie die Angaben.

Im Bereich *Aufgabe* sehen Sie die aktuelle Anzahl der Aufgaben im Aufgabenheft, Sie können zu einer Aufgabendefinitionsseite wechseln und neue Aufgaben an das Ende der Aufgabenhefts anfügen bzw. die letzte Aufgabe des Aufgabenhefts löschen. Wie Aufgaben definiert werden können, erfahren Sie im nächsten Kapitel.

4 Aufgabenerstellung

WebAssign sieht vor, dass alle Aufgaben Musterlösungen, Korrekturseiten und Quittungen als HTML-Seiten angelegt werden. Bilder, Grafiken oder Aufgaben, die mit einem beliebigen Programm geschrieben wurden, werden als Kursressourcen¹ über einen Link eingebunden.

4.1 Verschiedene Aufgabentypen

Insgesamt unterstützt WebAssign drei Aufgabentypen, die jeweils verschiedene Bewertungsverfahren haben:

- Multiple-Choice-Aufgaben, Zuordnungsaufgaben, Begriffs- und numerische Aufgaben werden automatisch korrigiert und bewertet
- Frei-Text-Aufgaben werden manuell korrigiert und bewertet
- Aufgaben mit Vortest (Pre-Testing) werden zusätzlich noch manuell korrigiert und bewertet

1. Kursressourcen werden im Handbuch für Betreuer beschrieben.

tet. Beispiel: Aufgaben zur Programmierung. Programme können von den Studierenden eingesendet werden; WebAssign übersetzt die Programme und anhand von vorbereiteten Testfällen werden die Programme zusätzlich noch auf ihre semantische Korrektheit überprüft; anschließend werden sie von einem Korrektor manuell bewertet. Der Studierende kann unmittelbar nach dem Einsenden einer Lösung Zugriff auf die Vortestausswertung bekommen und die Hinweise zum Überarbeiten seiner Lösung verwenden.

4.2 Aufgabenstruktur

Eine Aufgabe umfaßt stets die Aufgabenseite, die Korrektorschablone und die Quittungsseitenschablone:

- Eine Aufgabenseite besteht aus dem Aufgabentext und ihren eventuellen Teilaufgaben sowie aus einem oder mehreren Eingabefeldern, in die die Studierenden später die Lösungen eintragen sowie den entsprechenden Einsendebutton, mit denen die Studierenden ihre (Teil-)Aufgaben einsenden.
- Die Korrekturseite ist eine Schablone, die festlegt, welche äußere Form eine Korrektur hat. Sie enthält einen vorgefertigten Rahmen für die Daten des Studierenden (Name, Matrikelnummer, etc.) und die studentische Lösung. Später schreiben die Korrektoren ihre Kommentare in die mit Daten gefüllte Schablone und tragen die erreichten Punkte ein (siehe *Handbuch für Korrektoren*).
- Die Quittungsseite ist eine Schablone zur Erzeugung von Quittungen. Die Studierenden erhalten eine Quittung als sofortige (Online-)Bestätigung, nachdem sie ihre Lösung eingesendet haben. Sie enthält i. Allg. die eingesendete Lösung (siehe *Handbuch für Studierende*) und ggf. die Auswertung der Vortests.

Eine Aufgabe umfaßt wahlweise das Bewertungsmodul (ältere Bezeichnung: Korrekturmodul), ein oder mehrere Selbsttestmodule (ältere Bezeichnung: Vorkorrekturmodul), Hinweise für Korrektoren und die Musterlösung:

- Für automatisch auswertbare Aufgaben muss ein Bewertungsmodul angegeben werden, für solche mit Pre-Testing ein Selbsttestmodul.
- Mit den Hinweisen für die Korrektoren können Sie die Korrektoren über Besonderheiten der Aufgabe informieren, z.B. über den Schwerpunkt der Korrektur, erwartete Fehler etc. Diese Hinweise werden in die Korrekturseiten eingeblendet und stehen damit den Korrektoren zur Verfügung.
- Musterlösungen sind von Ihnen vorbereitete Lösungen, die die Studierenden nach Bearbeitungsende einsehen können. Anhand der Musterlösung können Studierende, die Schwierigkeiten mit der Lösung hatten, diese nachvollziehen.

4.3 Aufgabenerstellung

Um das Grundgerüst einer Aufgabe anzulegen oder um eine bestehende Aufgabe zu editieren, können Sie den Link der Aufgabe, z.B. *Aufgabe <n>* in der Aufgabenhefte/Aufgaben-Übersicht (siehe Abb. 3) auswählen, oder in der Aufgabenheftverwaltung die zu bearbeitende Aufgabe ansprechen (siehe Abb. 5). Sie gelangen in jedem Fall zu der *Aufgabenerstellungsseite*.

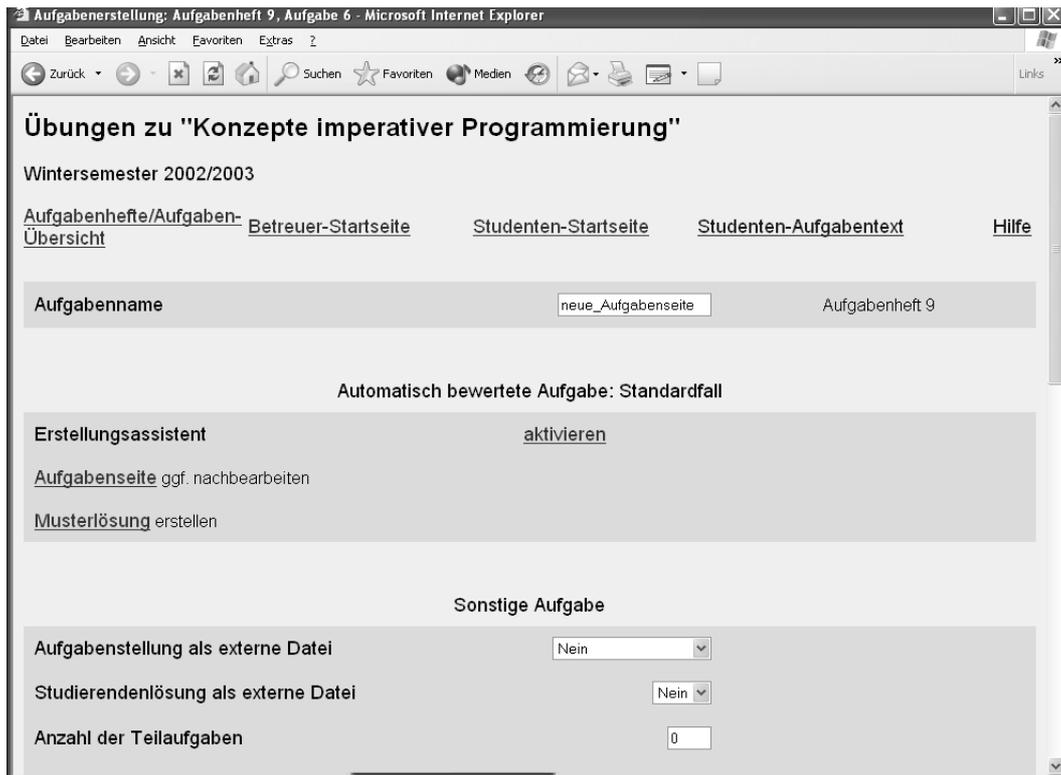


Abb. 6 Aufgabenerstellungseite

Auf dieser Seite können alle Informationen zu einer einzelnen Aufgabe angesehen und verändert werden. Hier können Sie der Aufgabe einen sprechenden Namen geben - standardgemäß werden die Aufgaben durchnummeriert und heißen *Aufgabe 1*, *Aufgabe 2*, etc.

Wenn Sie eine automatisch bewertete Aufgabe anlegen wollen, so sollten Sie den Erstellungsassistenten verwenden (siehe Abschnitt 4.4). Klicken Sie dafür auf den Link *aktivieren*. Der Erstellungsassistent erzeugt die Aufgabenseite und die Schablonen für die Quittung und die Korrektur. Das Bewertungsmodul wird automatisch eingetragen und richtig konfiguriert. Die studentische Musterlösung muss in jedem Fall noch selber erstellt werden. Nach Anklicken des Links *Musterlösung* wird zu der entsprechenden Funktionalität der Aufgabenerstellungseite gesprungen. Wollen Sie die vom Erstellungsassistenten erzeugte Aufgabenseite noch nachbearbeiten, so klicken Sie den Link *Aufgabenseite* an.

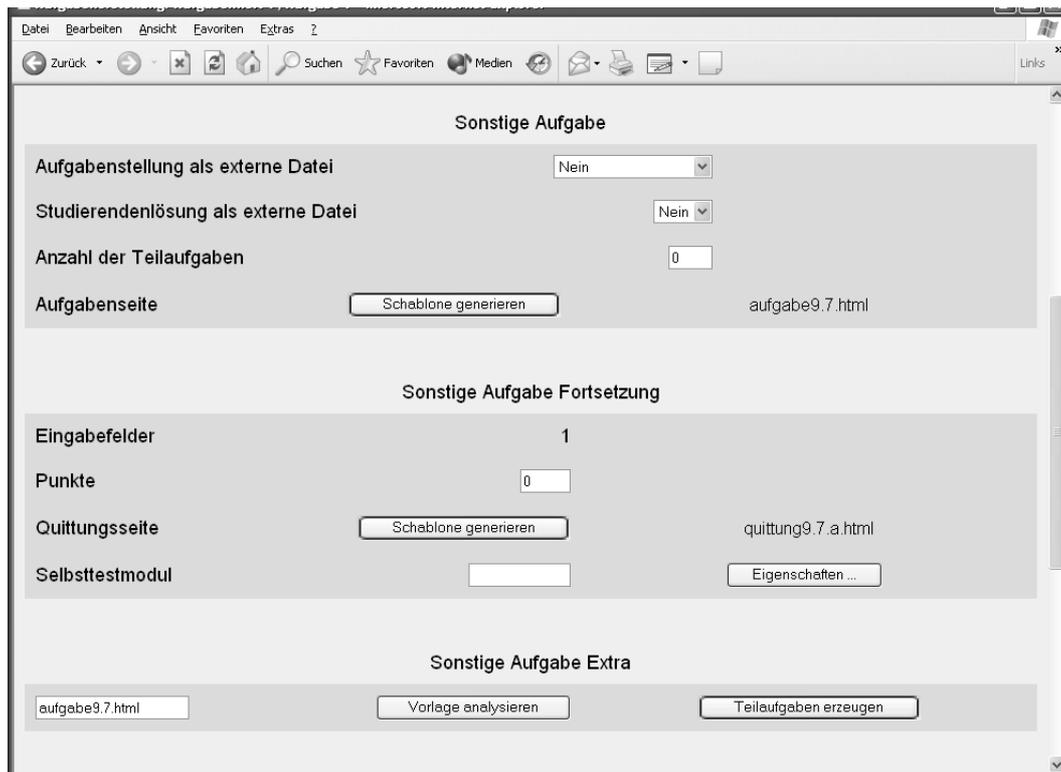


Abb. 7 Aufgabenerstellungseite, Sonstige Aufgabe

Wenn Sie eine nicht automatisch bewertete Aufgabe erstellen wollen, so müssen Sie die Aufgabenseite und die Schablonen für die Quittung und die Korrektur selber erzeugen. Dies geschieht im Abschnitt *Sonstige Aufgabe* der Aufgabenerstellungseite. (siehe Abb. 7)

Über eine Combobox können Sie angeben, ob die Aufgabenstellung als externe Datei, z.B. als pdf- oder Word-Dokument angeboten werden soll. Sie haben die Wahl zwischen: *Nein*, *Ja(pdf)*, *Ja(doc)* und *Ja(anderes Format)*. Auf der Aufgabenseite wird in Abhängigkeit der Auswahl ein Link für die Aufgabenstellung im externen Format generiert. Damit der Studierende später über diesen Link Zugriff auf die Aufgabenstellung bekommt, muss die entsprechende Datei unter dem Namen $\langle \text{AufgabenheftNr} \rangle_ \langle \text{Aufgabennr} \rangle .pdf$, $\langle \text{AufgabenheftNr} \rangle_ \langle \text{Aufgabennr} \rangle .doc$ oder $\langle \text{AufgabenheftNr} \rangle_ \langle \text{Aufgabennr} \rangle$ (für anderes Format) als Kursressource in das WebAssign-System geladen werden. Beispiel: Ihre Aufgabenstellung liegt digital in der Datei Aufgabe.pdf vor. Sie wollen diese Aufgabenstellung als 7. Aufgabe im 9.Aufgabenheft anbieten. Dann benennen Sie die Datei um in 9_7.pdf und lade diese als Kursressource in das WebAssign-System.

Über die Combobox *Studierendenlösung als externe Datei* bestimmen Sie, ob der Studierende seine Lösung in ein vorgesehene HTML-Eingabefeld eintragen muss, oder ob er diese in einem externen Format, wie z.B. pdf, doc o.ä. hochladen kann. Je nach Auswahl werden bei der Generierung der Aufgabenseite entsprechende Eingabemöglichkeiten vorgesehen.

Wenn Sie mehrere Teilaufgaben verwenden wollen, so müssen Sie deren Anzahl eintragen. Nach dem Generieren der Aufgabenseite passt sich dann die Aufgabenerstellungseite entsprechend an.

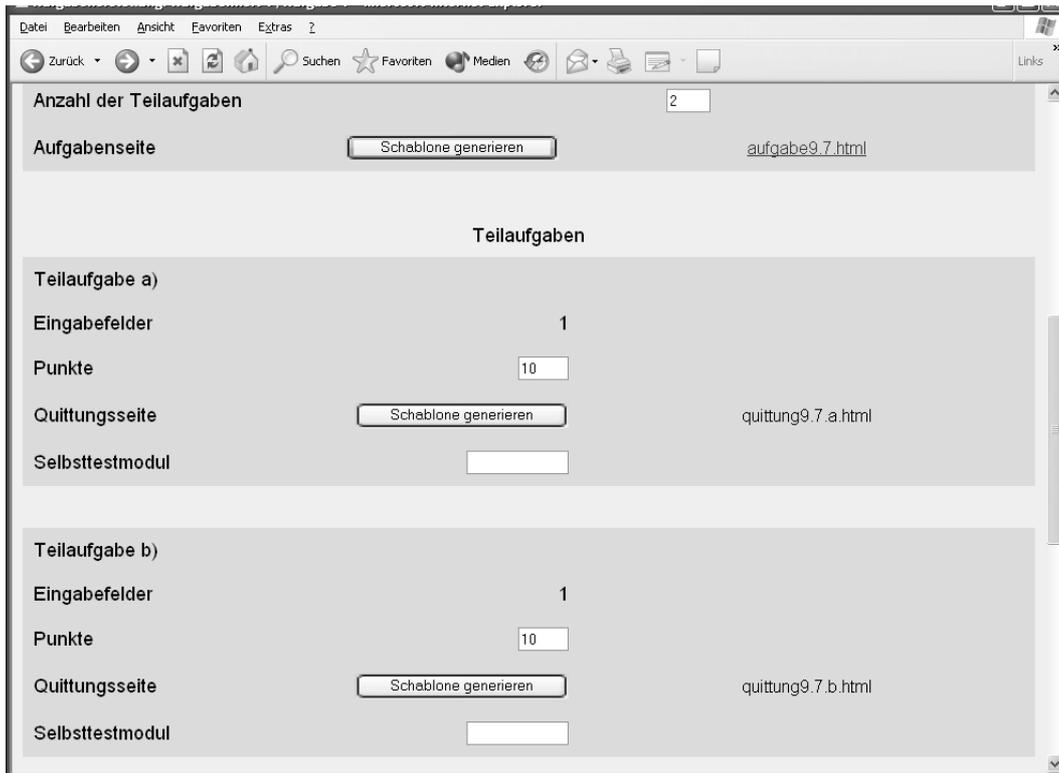


Abb. 8 Aufgabenerstellungsseite für mehrere Teilaufgaben

Im Abschnitt *Sonstige Aufgabe Fortsetzung* bzw. in den Teilaufgabenabschnitten haben Sie die Möglichkeit, die Punktzahlen und ggf. die Selbsttestmodule festzulegen.

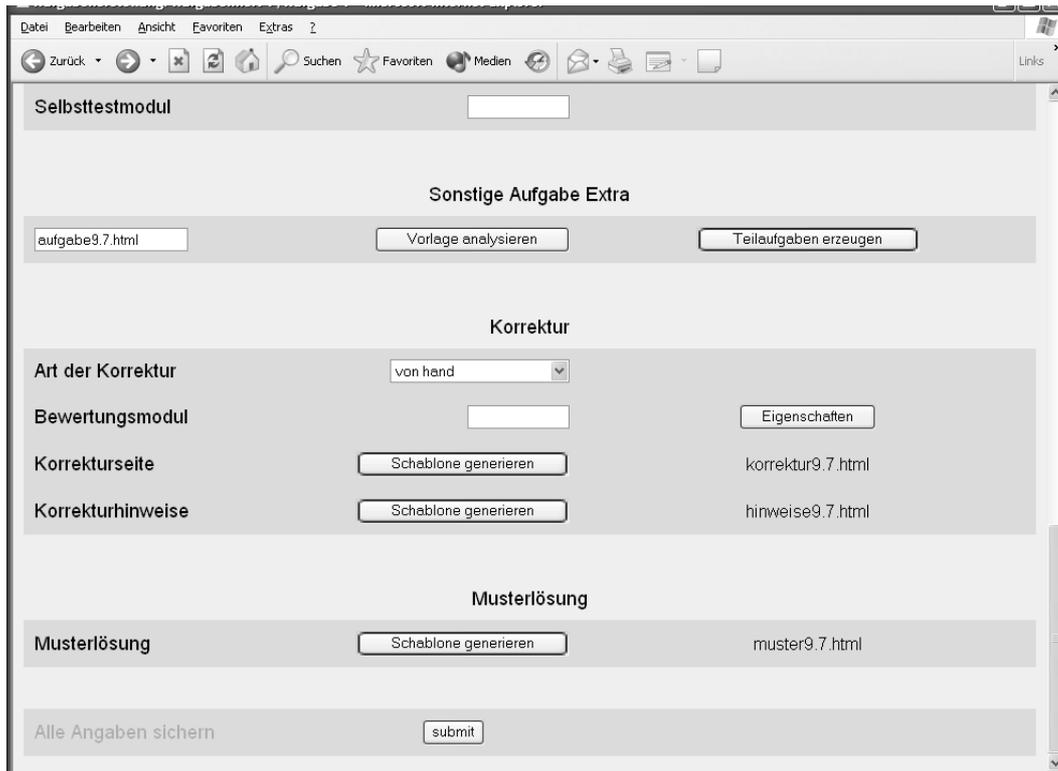


Abb. 9 Aufgabenerstellungsseite, Bereich Extra, Korrektur und Musterlösung

Der Abschnitt *Sonstige Aufgabe Extra* wird benötigt, wenn neue Eingabefelder oder Teilaufgaben über Vorlagen erzeugt werden sollen. Näheres erfahren Sie im Abschnitt 4.5.2.

Im Bereich *Korrektur* geben Sie über eine Combobox die Art der Korrektur an: *von hand*, *automatisch* oder *automatisch + von hand* und *keine*.

- Im Fall einer rein manuellen Korrektur wählen Sie als *Art der Korrektur* den Eintrag *von hand* aus.
- Im Fall der automatischen Korrektur wählen Sie als *Art der Korrektur* den Eintrag *automatisch* aus und geben unter *Bewertungsmodul* das entsprechende Bewertungsmodul an - sofern Sie nicht den Aufgabenerstellungsassistenten für automatisch auswertbare Aufgaben nutzen, der Ihnen viel Arbeit abnimmt (siehe Abschnitt 4.4).
- Bei Aufgaben mit Pre-Testing wählen Sie als Korrekturmodus *automatisch + von hand* aus und spezifizieren zu den Teilaufgaben das oder die entsprechende(n) Selbsttestmodul(e).
- Bei Aufgaben, für die es keine Punkte gibt, wie z.B. bei Selbsttestaufgaben wählen Sie als Korrekturart *keine* aus, unabhängig davon, ob Sie Bewertungsmodule oder Selbsttestmodule spezifiziert haben.

Wie schon erklärt, gehören zu einer Aufgabe die verschiedenen HTML-Seiten für Aufgabentext, Musterlösung, Hinweise für Korrektoren, die Korrekturseite sowie Angaben zur Aufgabenstruktur, d.h. den Teilaufgaben mit den zugeordneten Punkten.

Zu allen HTML-Seiten kann man über den Button *generieren* eine einfache Standardversion erzeugen. Die Standardversion der Korrekturseiten- und Quittungsschablonen können Sie unmittelbar verwenden.

Wahlweise nutzen Sie das vom Betreuer bereits angelegte und speziell angepasste Aufgaben-

gerüst oder Sie nutzen für automatisch auswertbare Aufgaben den Aufgabenerstellungsassistenten (siehe Abschnitt 4.4).

Wie der Aufgabentext schließlich für die Studierenden aussieht, sehen Sie, wenn Sie über die Links *Studenten-Startseite* oder *Studenten-Aufgabentext* das Layout der Aufgabe kontrollieren.

4.4 Der Aufgabenerstellungs-Assistent für automatisch auswertbare Aufgaben

Derzeit können mit dem Aufgabenerstellungs-Assistenten Aufgaben der folgenden Typen erstellt werden:

- Einfachauswahl,
- Mehrfachauswahl,
- Zuordnung,
- Zahl- oder numerische Aufgaben und
- Begriffe.

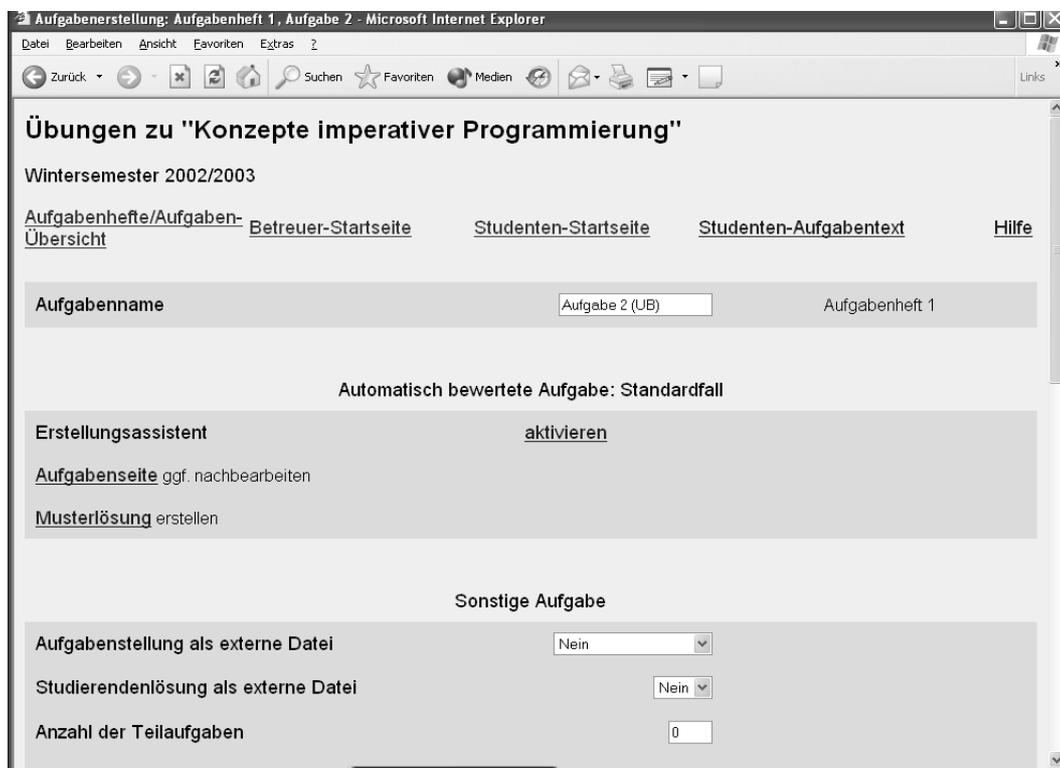


Abb. 10: Generieren neuer Aufgaben über den Aufgabenerstellungs-Assistent.

Um eine Aufgabe zu erstellen:

- rufen Sie den Aufgabenerstellungs-Assistenten auf,
- fügen Aufgaben des gewünschten Typs hinzu,
- füllen die nötigen Felder und
- wählen Aufgabenseite sichern aus.

Mit der Aufgabe werden zugleich die Korrekturschablone, die Quittungsseite und die Musterlösung erzeugt sowie die Art der Korrektur spezifiziert. Ändern Sie hier nichts. Sie starten den Aufgabenerstellungsassistent über den Link *Aufgabenerstellungs-Assistent* auf der Seite *Aufgabenerstellung* in der *Aufgabenverwaltung* (siehe Abb. 10 im oberen Bereich rechts).

Der Aufgabenerstellungsassistent wurde entwickelt, um die in der FernUniversität verwendeten Lotsebögen im Rahmen von WebAssign anbieten zu können. Das Lotsesystem weist Elemente wie den Seitentitel auf, die Sie ignorieren können, wenn Sie kein Lotse-konformes Layout¹ benötigen. Auch das Anlegen weiterer Aufgaben, sogenannte „Unteraufgaben“, in einer Aufgabe ist eine spezifische Lotse-Eigenschaft. Wenn Sie nicht Lotse-konform arbeiten, legen Sie für jede Aufgabe eine eigene Lotseseite an.

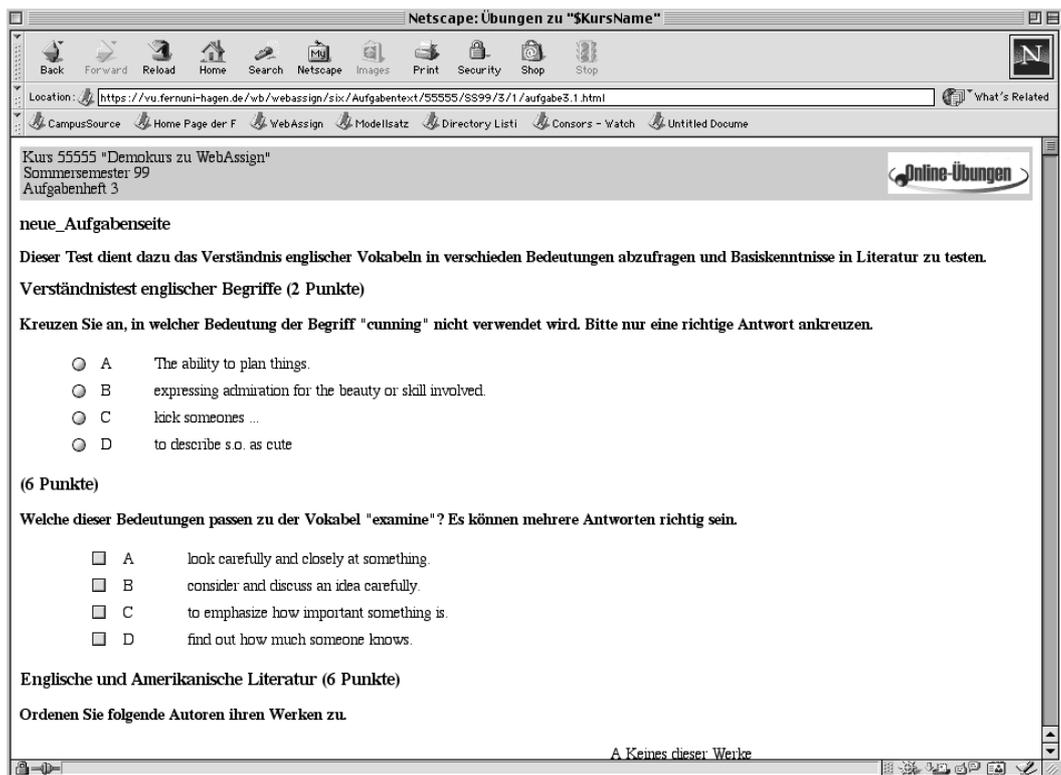


Abb. 11: Lotse-konformes Layout: Multiple-Choice mit Einfach- und Mehrfachauswahl (aus Studierendensicht).

1. Eine Lotseseite entspricht einer WebAssign Aufgabe, einem Lotse-Seitentitel dem Aufgabennamen.

Abb. 12: Multiple-Choice mit Einfachauswahl (aus Autorensicht).

Wenn Sie automatisch auswertbare Aufgaben erstellen möchten, ohne das Lotseformat zu verwenden, verzichten Sie auf den Seitentitel und den Einführungstext und fügen Sie genau eine Unteraufgabe pro Aufgabe ein. Dieses Vorgehen ermöglicht es Ihnen, für die automatisch auswertbaren Aufgabentypen, Kommentare in Abhängigkeit von der erreichten Punktzahl vorzusehen. (WebAssign ohne den Aufgabenerstellungsassistenten sieht keine allgemeinen Kommentare zu den studentischen Lösungen vor, alle Kommentare müssen von den Korrekturen geschrieben werden.)

Für Lotse-konforme Aufgaben füllen Sie zunächst die folgenden Felder aus (siehe Abb. 13).

<i>Aufgabenname</i>	Sie können hier einen spezifischen Namen eintragen, z.B. „Fragen zur Anatomie“. Standardgemäß werden die Aufgaben durchnummeriert und es wird als Name der Aufgabe Aufgabe<n> eingetragen.
<i>Globale Aufgabenstellung</i>	Geben Sie hier den Einführungstext an, der sich auf alle Unteraufgaben der Lotseseite bezieht.
<i>Gesamtbewertung Punkteintervall und Kommentar</i>	Sie können in Abhängigkeit von der erreichten Gesamtpunktzahl Kommentare, z.B. „gut“ oder „Sie sollten Kapitel 3 nochmals durcharbeiten“, ausgeben. Dazu tragen Sie unter <i>Punkteintervall</i> das gewünschte Intervall, z.B. „0 - 10“ und unter <i>Kommentar</i> den dazugehörigen Kommentar ein. Punkteintervalle gelten für die ganze Lotseseite.

*Korrekturzeitpunkt:
Korrektur sofort (online-Selbsttest)*

Hier wird festgelegt, wann die Korrektur erfolgen soll:
Bei der Auswahl *Korrektur sofort (online-Selbsttest)* bekommen die Studierenden das Ergebnis der Korrektur sofort nach der Einsendung angezeigt.

Korrektur nach Einsendeschluss

Bei der Auswahl *Korrektur nach Einsendeschluss* wird die Korrektur erst beim Einsendeschluss erzeugt und ist erst nach dem Bearbeitungsende für die Studierenden zugänglich.

Abb. 13: Der Aufgabenerstellungs-Assistent: Angaben, die sich auf die gesamte Aufgabe beziehen.

Als nächstes fügen Sie Aufgaben des gewünschten Typs hinzu. Wählen Sie aus dem Pop-Up Menü den Typ aus und bestätigen Sie die Wahl mit dem *generieren* Button (siehe Abb. 13 oben links). Wenn Sie nicht Lotse-konform arbeiten, fügen Sie nur eine Aufgabe hinzu.

Unabhängig vom Aufgabentyp müssen Sie die folgenden Felder (siehe Abb. 11) ausfüllen.

Aufgabentitel z.B. „Aufgabe zur Geographie von La Palma“

lokale Aufgabenstellung Geben Sie den Aufgabentext, die Frage, hier ein, z.B. „In welcher Klimazone liegt die Insel?“

Punkte

Geben Sie an, wie viele Punkte die Studierenden mit dieser Aufgabe maximal erreichen können. WebAssign kann nur ganzzahlige Punktzahlen vergeben, achten Sie daher darauf, **dass die Gesamtpunktzahl ein ganzzahliges Vielfaches der Anzahl der richtigen Antwortalternativen ist.**

lokaler Kommentar

Sie können den Studierenden hier zusätzliche Informationen geben, wie z.B. „Diese Aufgabe bezieht sich auf den Stoff von Kapitel 4“. Dieser Kommentar wird in der Lösung angezeigt.

Wenn Sie in den auszufüllenden Feldern und in den Aufgabentexten Umlaute verwenden, denken Sie bitte daran, die Umlaute in HTML zu codieren.

ä	ä
ö	ö
ü	ü
ß	ß
Ä	Ä
Ö	Ö
Ü	Ü

4.4.1 Multiple Choice mit Einfach- und Mehrfachauswahl

Bei Multiple Choice Aufgaben werden zwei Arten unterschieden:

- die Einfachauswahl, d.h. 1 aus n. Hier ist genau einer der angebotenen Antwortvorschläge richtig (siehe Abb. 12).
- die Mehrfachauswahl, d.h. m aus n. Hier kann es mehrere richtige Antwortvorschläge geben.

Es werden standardgemäß jeweils 5 Antwortmöglichkeiten generiert. Wählen Sie *<n>Alternativen hinzufügen*, wenn Sie mehr Antwortmöglichkeiten benötigen, wählen Sie *<n> Alternativen löschen*, wenn Sie weniger Antwortmöglichkeiten benötigen (siehe Abb. 12). Tragen Sie den Aufgabentitel, die lokale Aufgabenstellung und die Antwortmöglichkeiten ein. Um die automatische Korrektur zu ermöglichen, markieren Sie die richtige Alternative (Einfach-Auswahl eine Antwort, Multiple-Choice alle richtigen Alternativen) (siehe Abb. 14).

Aufgabenname		Sonnets
Globale Aufgabenstellung		Im folgenden geht es um die verschiedenen Arten von Sonnets.
Aufgabenart	Mehrfachauswahl	
Aufgabentitel	Italian Sonnets	
lokale Aufgabenstellung (optional)	Was sind die Bestandteile eines Italian Sonnets? Mehrere Antworten sind möglich.	
Alternative	richtig	Text
1	<input type="checkbox"/>	3 Quartette
2	<input checked="" type="checkbox"/>	2 Quartette
3	<input type="checkbox"/>	1 Zweizeiler
4	<input type="checkbox"/>	2 Zweizeiler
5	<input checked="" type="checkbox"/>	2 Terzette

Abb. 14: Die richtigen Alternativen werden markiert.

Unter die Antwortmöglichkeit tragen Sie in das Feld *Punkte*, die Gesamtzahl der Punkte für diese Unteraufgabe ein (siehe Abb. 7).

Unter *lokaler Kommentar* tragen Sie den Text ein, den alle Studierenden, unabhängig von den erzielten Punkten, als zusätzliche Information für diese Unteraufgabe sehen sollen, wenn sie die Korrektur erhalten, z.B. „Diese Aufgabe bezieht sich auf Abschnitt 5.“

Im unteren Teil können Sie den Studierenden in Abhängigkeit von der erreichten Punktzahl aller Unteraufgaben dieser Aufgabe bzw. des Lotsebogens ein Feedback zukommen lassen (siehe Abb. 7). Bestimmen Sie schließlich noch, ob die Aufgabe direkt beim Einsenden korrigiert werden soll oder ob die Aufgabe erst zum Einsendeschluss korrigiert werden soll, d.h. ob die Ergebnisse den Studierenden sofort nach dem Einsenden der Lösung mitgeteilt werden soll, oder ob sie diese erst nach Einsendeschluss erhalten sollen.

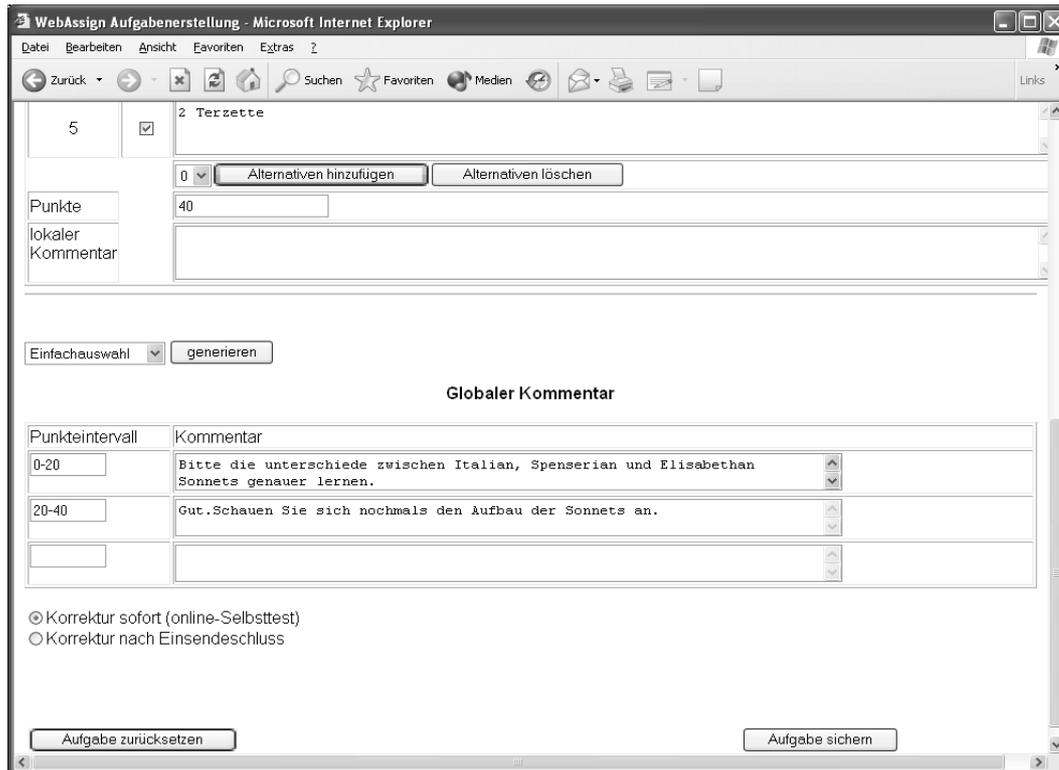


Abb. 15: Sie können Kommentare zu Punkteintervallen eintragen.

Sichern Sie die Aufgabe über den Button *Aufgabe sichern* oder löschen Sie die Aufgabenangaben über den Button *Aufgabe zurücksetzen* (siehe Abb. 15).

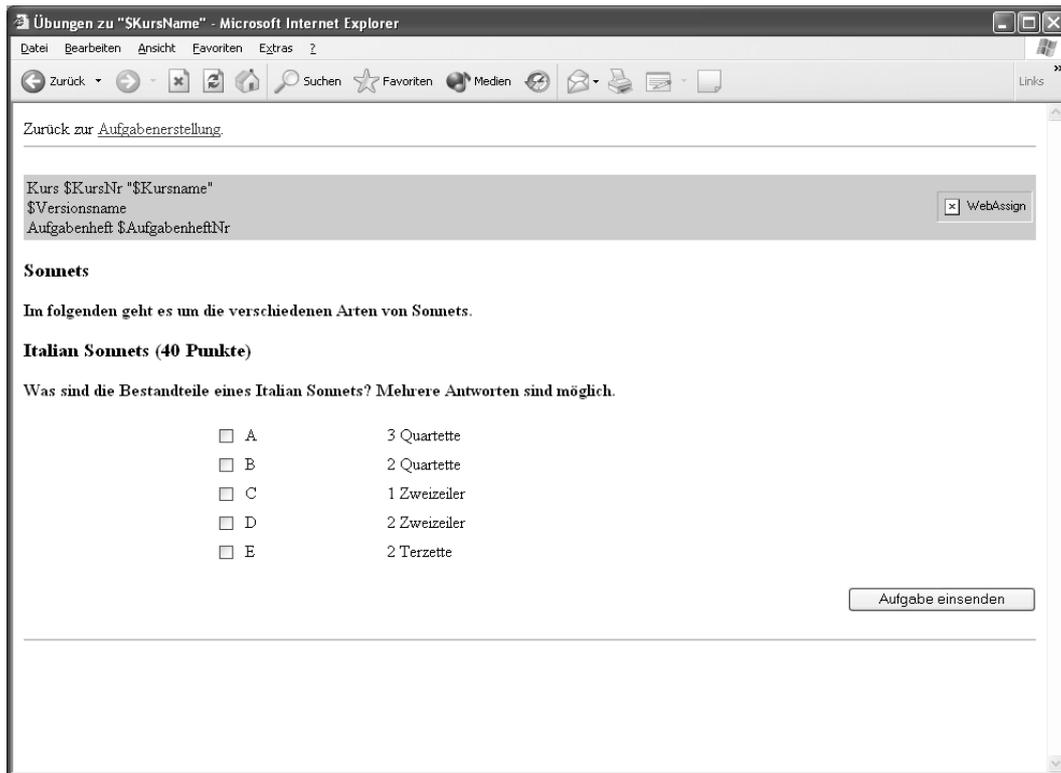


Abb. 16: Nach dem Sichern der Aufgabe wird das Layout angezeigt.

Nach dem Sichern der Aufgabe wird Ihnen das Layout der Aufgabe zur Kontrolle angezeigt (siehe Abb. 16). Dabei werden den von WebAssign oder vom Kursbetreuer vorkonfigurierten Variablen noch keine Werte zugewiesen. Die zugewiesenen Werte sehen Sie nur aus Studierendensicht (siehe Abb. 17). Mit dem Link *Zurück zur Aufgabenerstellung* (oben auf der Seite) kommen Sie zurück und können die Aufgabe weiter editieren.

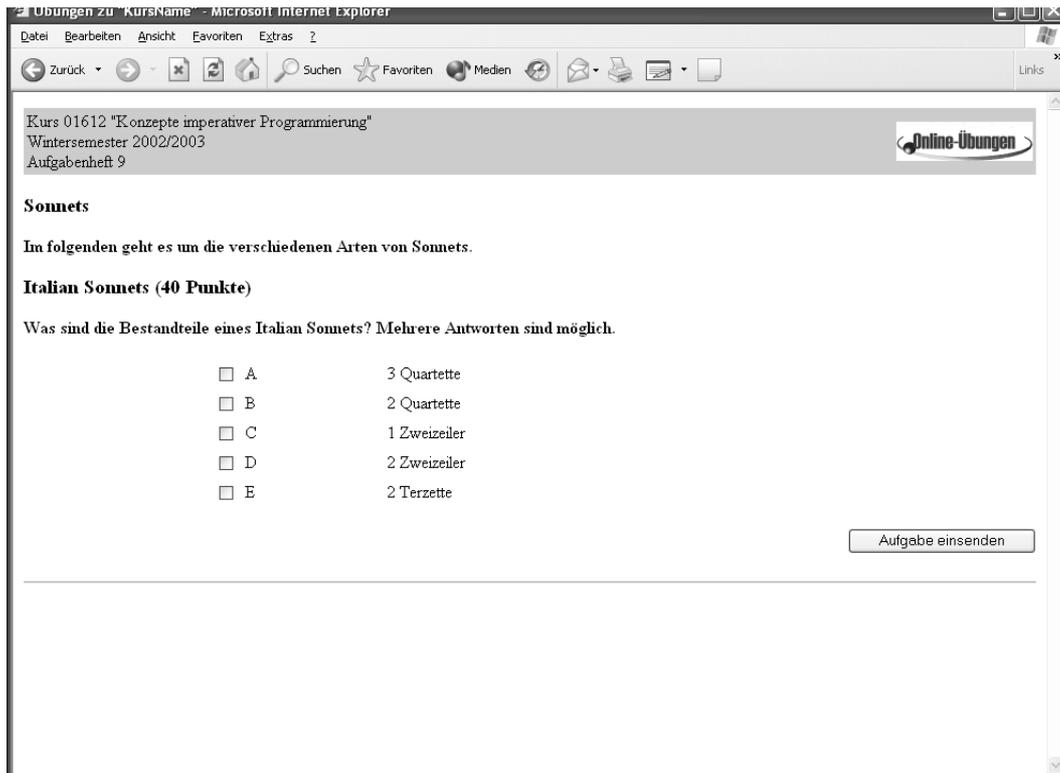


Abb. 17: Layout der Aufgabe aus Sicht der Studierenden

4.4.2 Zuordnungsaufgaben

Bei einer Zuordnungsaufgabe (siehe Abb. 18) sollen die Studierenden die jeweils richtige Antwort ankreuzen, wobei Fragen und Antworten in Matrixform angeboten werden. Im ersten Schritt fügen Sie eine Zuordnungsaufgabe über die Combobox *Zuordnung* und den Button *Aufgabe hinzufügen* hinzu. Geben Sie den Einführungstext, den Aufgabentitel sowie den Aufgabentext an und fügen Sie entsprechend viele Fragen mit *<n> Fragen hinzufügen* hinzu - standardgemäß wird eine Frage erzeugt. Über *<n> Fragen löschen* können Sie Fragen entfernen. Dabei können Sie zur Zeit nicht spezifizieren, welche Fragen gelöscht werden sollen, sondern es werden die letzten *<n>* Fragen gelöscht. Geben Sie die Fragen und die Antwortmöglichkeiten vor und kreuzen Sie die richtigen Antworten zur automatischen Auswertung an. Sichern Sie die Aufgabe über den Button *Aufgabenseite sichern*. Nach dem Sichern erhalten Sie den Preview der Aufgabe zur Kontrolle (siehe Abb. 19). Mit dem Link *Zurück zur Aufgabenerstellung* kommen Sie zurück und können die Aufgabe weiter editieren. Mit dem Button *Aufgabe löschen* können Sie die zuletzt hinzugefügte Aufgabe löschen.

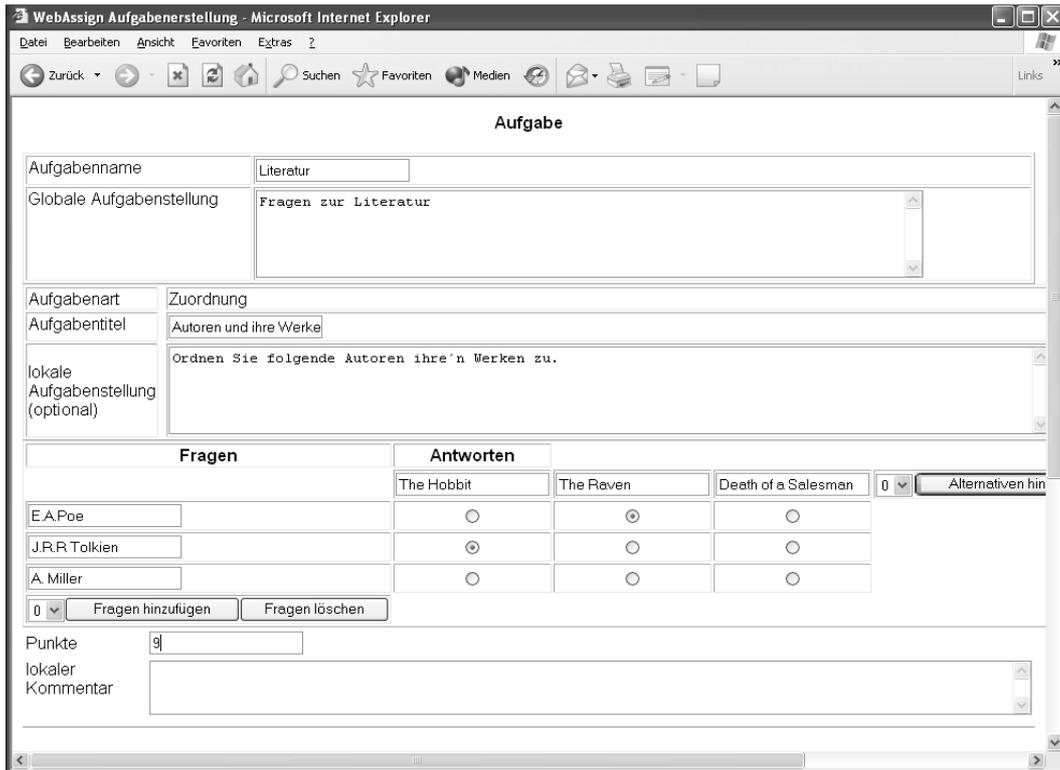


Abb. 18: Zuordnungsaufgabe mit 3 Fragen

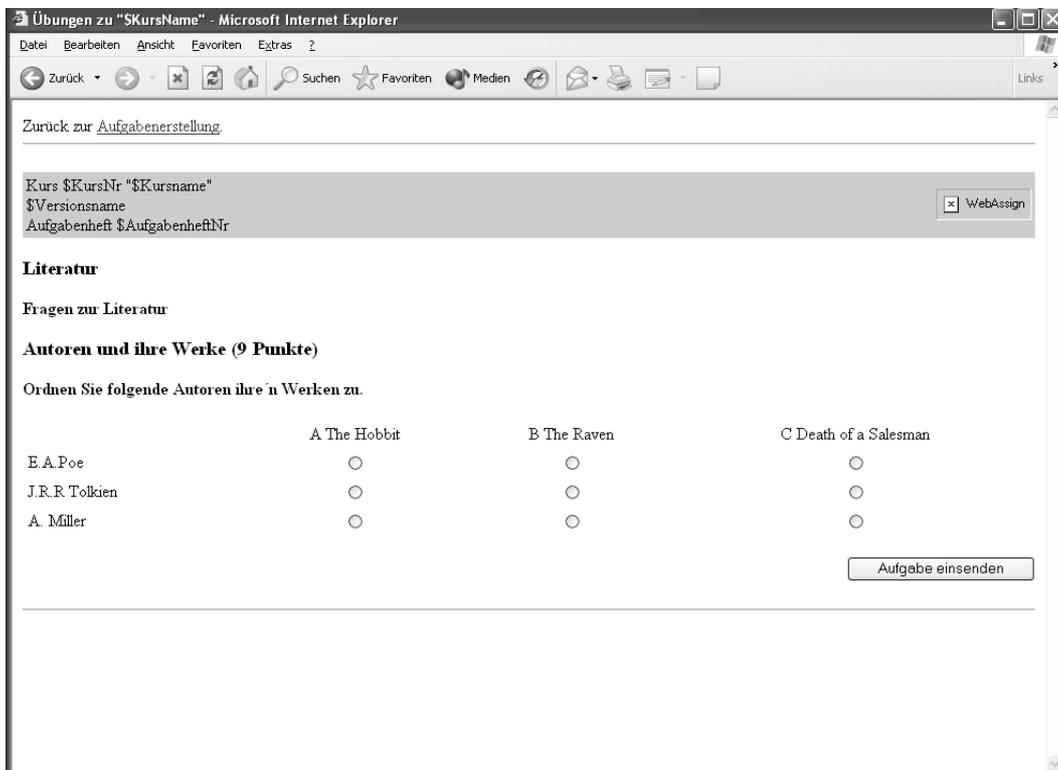


Abb. 19: Preview einer Zuordnungsaufgabe

4.4.3 Zahl-Aufgaben

Eine numerische Aufgabe oder Zahl-Aufgabe ist eine Aufgabe, bei der die Antwort aus einer Zahl besteht (siehe Abb. 20).

Aufgabenname	<input type="text"/>
Globale Aufgabenstellung	<div style="border: 1px solid gray; padding: 2px;">Dies ist der Einführungstext</div>
Aufgabenart	Zahlen
Aufgabentitel	<input type="text" value="Zahlaufgabe"/>
lokale Aufgabenstellung (optional)	<div style="border: 1px solid gray; padding: 2px;">Dies ist eine numerische Aufgabe. Wieviel Prozent sind die Studentenzahlen gefallen?</div>
Minimum	<input type="text" value="8.1"/>
Maximum	<input type="text" value="8.1"/>
Punkte	<input type="text" value="5"/>
lokaler Kommentar	<div style="border: 1px solid gray; padding: 2px;">Kommentar</div>

Einfachauswahl

Globaler Kommentar

Punkteintervall

Abb. 20: Numerische (Zahl-) Aufgabe.

Im ersten Schritt fügen Sie über die Combobox *Zahlaufgabe* und den Button *generieren* eine Aufgabe hinzu. Tragen Sie den Aufgabentext und die Punkte ein. Erwarten Sie als Antwort genau einen Wert, geben Sie in die Felder *Minimum* und *Maximum* denselben Wert ein. Kann die Zahl in einem Intervall liegen, geben Sie die obere und untere Grenze für die automatische Auswertung an (siehe Abb. 20).

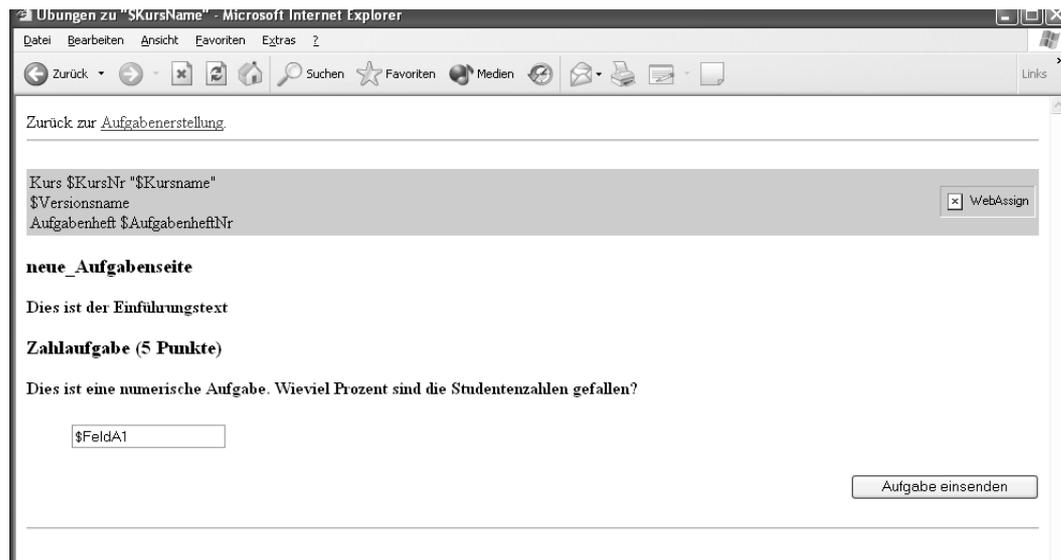


Abb. 21: Vorschau der numerischen Aufgabe

Sichern Sie die Aufgabe über den Button *Aufgaben sichern*. Nach dem Sichern erhalten Sie den Preview der Aufgabe zur Kontrolle (siehe Abb. 21). Da bei der Definition kein Aufgabennamen angegeben wurde, ergänzt WebAssign diesen automatisch und wählt den Bezeichner *neue_Aufgabenseite*. Mit dem Link *Zurück zur Aufgabenerstellung* kommen Sie zurück und können die Aufgabe weiter editieren.

4.4.4 Begriffsabfrage

Bei einer Begriffsabfrage müssen die Studierenden einen oder mehrere Begriffe angeben (siehe Abb. 22). Zunächst erstellen Sie über die Combobox *Begriffe* und den Button *Aufgabe hinzufügen* eine Begriffsabfragen-Aufgabe. Fügen Sie über *<n> Begriffe hinzufügen* die benötigte Anzahl Begriffe hinzu. Über *<n> Begriffe löschen* löschen Sie vom Ende her *<n>* Begriffe. Tragen Sie nun den Aufgabentext und die Anzahl der Begriffe und der abgefragten Begriffe ein (siehe Abb. 22). Der Unterschied zwischen *Anzahl abgefragter Begriffe hinzufügen* und *Begriffe hinzufügen* wird am besten an einem Beispiel deutlich:

Lautet die Aufgabe: Nennen Sie drei von neun Planeten! So ist „drei“ die Anzahl der abgefragten Begriffe und „neun“ die Anzahl der (möglichen) Begriffe. Wenn die Studierenden mehr als drei Planeten angeben sollen, so wählen Sie *Anzahl abgefragter Begriffe hinzufügen* und wenn Sie die Menge der möglichen Planeten auf zwölf erhöhen möchten, so wählen Sie *Begriffe hinzufügen*.

Zu einem Begriff können verschiedene Synonyme als gleichwertige Antworten angegeben werden, z.B. falls für deutsche Bezeichnungen auch gängige englische Bezeichnungen existieren. Wählen Sie *Synonyme <n> Synonyme hinzufügen* aus, um *n* Synonyme hinzuzufügen. Es werden stets für alle Begriffe gleichviel Synonyme angelegt. Lassen Sie zusätzliche, nicht benötigte Felder frei. Reduzieren Sie die Zahl der Synonyme, indem Sie *Synonyme <n> Synonyme löschen* auswählen.

The screenshot shows a web browser window titled 'webassign: Aufgabenerstellung - Microsoft Internet Explorer'. The main content is a form titled 'Aufgabe'. The form has several sections:

- Aufgabenname:** A text input field containing 'Synonyme'.
- Globale Aufgabenstellung:** A large empty text area.
- Aufgabenart:** A dropdown menu set to 'Begriffe'.
- Aufgabentitel:** An empty text input field.
- lokale Aufgabenstellung (optional):** A text area containing the instruction: 'Schreiben Sie in die unteren Felder die englischen Synonyme für Fell. Es können auch mehrere Antworten richtig sein.'
- Anzahl der abgefragten Begriffe:** A dropdown menu set to '1'.
- Table:** A table with two columns: 'Begriff' and 'Synonyme'. The 'Begriff' column contains 'coat', 'fur', and 'pelt'. The 'Synonyme' column is empty. There are buttons 'Begriffe hinzufügen' and 'Begriffe löschen' below the table.
- Übereinstimmung:** A text input field containing '100'.
- Punkte:** A text input field containing '20'.
- lokaler Kommentar:** An empty text area.

Abb. 22: Begriffsabfrage mit Synonymen

Unter dem Punkt *Übereinstimmung* können Sie angeben, inwieweit eine Tippfehlertoleranz ermöglicht wird. Dem verwendeten Algorithmus liegt zur Zeit jedoch ein recht schlichter Vergleich über die Anzahl der richtigen Buchstaben zugrunde, so empfehlen wir mit diesem Wert nur konservative Übereinstimmungen anzugeben.

Beispiel: *Hut* und *Hit* haben von 3 Buchstaben 2 gleiche und damit eine Übereinstimmung von $\frac{2}{3}$, obwohl ihre Semantik i.a. völlig unterschiedlich ist.

The screenshot shows a web browser window titled 'Übungen zu "\$KursName" - Microsoft Internet Explorer'. The main content is a task preview:

- Navigation:** A link 'Zurück zur Aufgabenerstellung'.
- Course Information:** A grey box containing 'Kurs \$KursNr "\$Kursname"', '\$Versionsname', and 'Aufgabenheft \$AufgabenheftNr'.
- Section Header:** 'Synonyme'.
- Points:** '(20 Punkte)'.
- Instruction:** 'Schreiben Sie in die unteren Felder die englischen Synonyme für Fell. Es können auch mehrere Antworten richtig sein.'
- Input Field:** 'a) \$FeldA1' with an empty text input field.
- Submit Button:** 'Aufgabe einsenden'.

Abb. 23: Preview einer Aufgabe mit Begriffsabfrage.

Sichern Sie die Aufgabe über den Button *Aufgabenseite sichern*. Nach dem Sichern erhalten Sie den Preview der Aufgabe zur Kontrolle (siehe Abb. 23). Mit dem Link *Zurück zur Aufgabenerstellung* kommen Sie zurück und können die Aufgabe weiter editieren. Mit dem Button *Aufgabe löschen* können Sie die Aufgabe löschen.

4.4.5 Sichern und Ändern der Aufgaben

Wenn Sie alle Informationen eingegeben haben, muss die Aufgabenseite gesichert werden. Dazu wählen Sie den Button *Aufgabenseite sichern* wie bei den einzelnen Aufgabentypen beschrieben. Sie erhalten eine Vorschau der fertigen Aufgabenseite. Neben der Aufgabenseite werden die notwendigen Konfigurationen für die automatische Korrektur, die Korrekturseite und die Quittungsseite erstellt und gespeichert. Um weitere Änderungen vorzunehmen, können Sie mit *Zurück zur Aufgabenerstellung* auf die vorher angezeigte Seite wechseln und anschließend nach dem Editieren über den Button *Aufgabenseite sichern* die Aufgabe erneut sichern. Wir raten davon ab, manuell editierte Aufgaben (wieder) mit dem Aufgabenerstellungsassistenten zu bearbeiten.

4.5 Frei-Text-Aufgaben

Frei-Text-Aufgaben sind Aufgaben, bei denen die Studierenden in ein oder mehrere Eingabefelder ihre Lösungen schreiben, z.B. frei formulierte Antworten wie Essays oder Programmcode. Dabei haben Sie zwei verschiedene Verfahren zur Auswahl: Sie können die Aufgabe so gestalten, dass die Studierenden ihre Lösungen in von Ihnen angelegte HTML-Eingabefelder schreiben oder kopieren (siehe Abschnitt 4.5.3) oder so, dass die Studierenden Dateien, die in beliebigen Programme erstellt wurden, auf den WebAssign-Server hochladen können (siehe Abschnitt 4.10).

Das prinzipielle Vorgehen ist folgendermaßen:

Für Frei-Text-Aufgaben generieren Sie die HTML-Seite für den Aufgabentext über die Aufgabenerstellungsseite. Wählen Sie in der Combox *Aufgabenstellung als externe Datei* „Nein“ und drücken dann den generieren-Button. Danach wird der Text *aufgabe <n.n>.html* zum Link, d.h. unterstrichen.

Wählen Sie den Link *Aufgabentext <n.n>.html* aus, um den Aufgabentext einzugeben. Um die Seite zu editieren, wählen Sie im Browser-Menü *Datei-Seite bearbeiten* bzw. *File-Edit Page* aus. Nachdem Sie den Aufgabentext¹ eingegeben haben, sichern und übertragen Sie die Seite auf den WebAssign-Server mit *Datei-Seite publizieren* bzw. *File-Publish* aus dem Browser-Menü. Schließen Sie anschließend den Composer und kehren Sie zur Aufgabenkonfiguration zurück. Danach weisen Sie WebAssign an, die (technischen) Teilaufgaben zu erzeugen (siehe Abschnitt 4.5.2) und überprüfen anschließend die Aufgabe, indem Sie sie aus Sicht der Studierenden und Korrektoren betrachten und bearbeiten.

4.5.1 Variablen in Aufgabentexten

Um die HTML-Aufgabenstruktur kennen zu lernen, publizieren Sie eine neu erzeugte Aufgabe, d.h. das Grundgerüst einer Aufgabe (siehe Abb. 24). Immer wiederkehrende Elemente, wie der

1. Wenn Sie den Netscape Composer verwenden, können Sie beim Erstellen des Aufgabentextes auch Umlaute verwenden.

Kursname oder variable Elemente, wie der Name des einsendenden Studierenden und dessen Lösung werden über Variablen in die HTML-Seiten der Aufgaben, Musterlösungen, Quittungen oder Korrekturseiten eingebunden. Variablen erkennen Sie daran, dass sie mit einem \$-Zeichen beginnen. Die Variable *\$Kursname* wird beispielsweise für den Kurs „Imperative Programmiersprachen“ durch eben diesen Kurstitel ersetzt. Ändern Sie keine Variablen auf den HTML-Seiten. Wir gehen in diesem Handbuch davon aus, dass die Variablen vom Betreuer gesetzt wurden, um ein einheitliches Erscheinungsbild der Aufgaben zu gewährleisten. Sind Sie Autor und Betreuer in Personalunion, lesen Sie bitte im *Handbuch für Betreuer* über Variablen nach.



Abb. 24: Grundgerüst einer Aufgabe.

Wird der Aufgabentext später von Studenten eingesehen, werden den Variablen die aktuellen Werte von WebAssign zugewiesen. Die Variablen auf der generierten Standardseite sind \$KursNr, \$Kursname, \$Versionsname und \$AufgabenheftNr (siehe Abb. 25).



Abb. 25: Publiziertes Grundgerüst einer Aufgabe.

Genauso wie Sie die Variablen nicht ändern sollten, sollten Sie auch die Texte wie *Kurs*, *Einsendeaufgaben*, *Aufgabenheft*, *Aufgabe <Nr>* nicht ändern.

In die Zeile unter *Aufgabe <Nr>* schreiben Sie den Aufgabentext, d.h. die Anweisungen für die Studierenden. Damit die Studierenden Ihre Lösungen eingeben können, legen Sie Eingabefelder an und damit Sie Ihre Lösungen einsenden können, legen Sie Einsendebutton an.

Benötigen Sie nur ein Eingabefeld, brauchen Sie nichts weiter zu tun. Benötigen Sie weitere

Eingabefelder, z.B. für Teilaufgaben, müssen Sie diese zusammen mit den zugehörigen Einsendebutton anlegen (siehe Abschnitt 4.5.3 und Abschnitt 4.5.4) sowie WebAssign anweisen, diese in die Datenbank anzulegen (siehe Abschnitt 4.5.2 und Abschnitt 4.12).

4.5.2 Aufgabe analysieren und Teilaufgaben erzeugen

Im Sinne von WebAssign unterscheidet man zwei Arten von Teilaufgaben, die logischen Teilaufgaben, die Sie als Autor zur Strukturierung der Aufgabe vornehmen und die technischen Teilaufgaben, die WebAssign pro Eingabebutton mit dem(n) zugehörigen Eingabefeld(ern) anlegt. In diesem Handbuch werden nur die technischen Teilaufgaben behandelt.

Sie müssen WebAssign anweisen, aus den von Ihnen angelegten Eingabefeldern und Einsendebutton die technischen Teilaufgaben zu erzeugen. Wir empfehlen, zuvor die Aufgabe über den Button *Vorlage analysieren* analysieren zu lassen (siehe Abb. 9), die Ergebnisse der Analyse anzuschauen (siehe Abb. 26) und gegebenenfalls die Aufgabe zu korrigieren. Zeigt die Analyse, dass der HTML-Text Ihren Erwartungen entspricht, lassen Sie die Teilaufgaben erzeugen. Wählen Sie dazu den Button *Teilaufgaben erzeugen* aus. WebAssign liegt nun in der Datenbank entsprechende Felder an.

Überprüfen Sie schließlich die Aufgabe, indem Sie dem Link *Studenten-Aufgabentext* folgen (siehe Abb. 12 oben rechts).

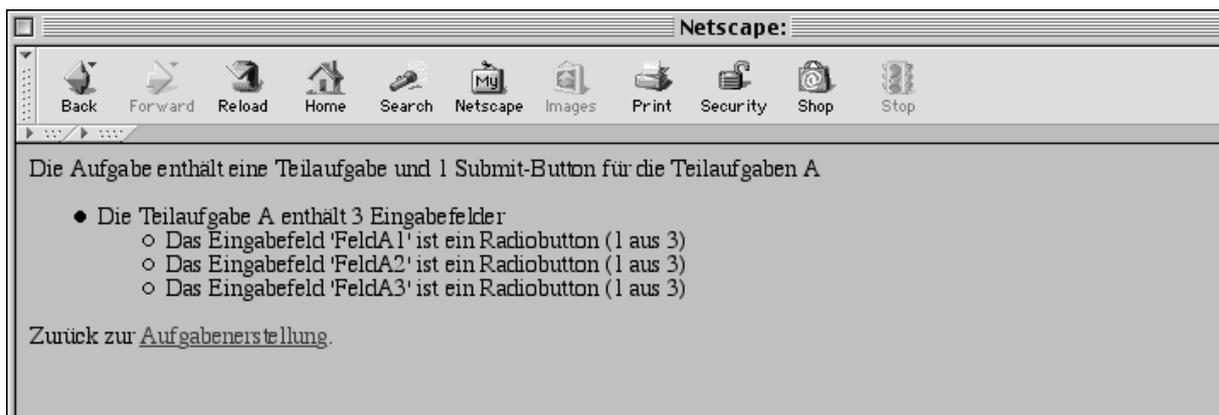


Abb. 26: Aufgabenanalyse.

4.5.3 Anlegen weiterer Eingabefelder

```
<HTML>
<HEAD>
  <TITLE>&Uuml;bungen zu "$Kursname"</TITLE>
</HEAD>
<BODY>
<FORM METHOD="POST" ACTION="$WebAssignServer/$Veranstaltername/Einsendung/$KursNr/$VersionsNr/$AufgabenheftNr/$AufgabenNr">
  <H1>
    Kurs $KursNr
  </H1>
  <H1>
    "$Kursname"
  </H1>
  <FONT SIZE="+1">$Versionsname</FONT> <FONT SIZE="+1">Einsendeaufgaben Aufgabenheft $AufgabenheftNr</FONT>
  <H2>
    Aufgabe 3 a)
  </H2>
  <BR>
  <INPUT TYPE="TEXT" NAME="FeldA1" VALUE="$FELDA1">
  <BR>
  <INPUT TYPE="submit" NAME="einsendenA" VALUE="Aufgabe $AufgabenNr einsenden">
  <P>
    <HR ALIGN="LEFT">
  </P>
</FORM>
</BODY>
</HTML>
```

Abb. 27: Der HTML-Text der generierten Aufgabe.

Öffnen Sie die HTML-Seite mit dem Aufgabentext im Netscape Composer¹. Wählen Sie im Browser-Menü unter *Communicator: Die Seite im Composer öffnen* aus. Das standardgemäß angelegte Eingabefeld erkennen Sie am ersten gelben Pfeil unter dem Text *Aufgabe 4 a)* (siehe Abb. 29). Der zweite gelbe Pfeil stellt hier den Einsendebutton dar. Klicken Sie ein Pfeilsymbol doppelt an, öffnet sich ein Fenster mit dem entsprechenden HTML-Text (siehe Abb. 29). Das Eingabe-Text-Feld hier wird durch

```
<textarea rows=20 cols=75 NAME="FeldA1" VALUE="Aufgabe einsenden">
```

spezifiziert. Dabei besagt TEXTAREA, dass es sich um ein Eingabefeld vom Typ TEXTAREA handelt, der NAME dient zur Identifikation und späteren Zuordnung der Eingabe zur (Teil-) Aufgabe. Mit den Attributen ROWS und COLS geben Sie, wenn Sie möchten, an, wie viele Zeichen das Eingabefeld breit und lang sein soll (siehe jede HTML - Referenz).

Im Wesentlichen werden Sie folgende Typen von Eingabefeldern nutzen (siehe Abb. 28). Weitere Informationen finden Sie beispielsweise unter <http://selfhtml.teamone.de/html/formulare/>

1. Wie Sie mit dem Composer arbeiten ist unter <http://home.netscape.com/eng/mozilla/4.0/handbook/comp.htm> erklärt. Diese Dokumentation ist nur auf Englisch vorhanden und hat einen Umfang von rund 45 Seiten.

eingabe.htm#bereiche.

<code><input type="text" name="FeldA1"></code>	definiert ein einzeiliges Eingabefeld (input = Eingabe). Bei einzeiligen Eingabefeldern können Sie die Anzeigelänge in Zeichen mit <i>size</i> = "...“ sowie die interne Feldlänge in Zeichen mit <i>maxlength</i> = "...“ bestimmen.
<code><input type="file" name="FeldA1"></code>	legt ein Feld und einen Button an, mit dem man Dateien auf den Server hochladen kann. Wählt man den Button an, erscheint eine Dateiauswahl-Box und der Name bzw. Pfad der ausgewählten Datei erscheint im Eingabefeld.
<code><textarea name="FeldA1" cols="50" rows="10"> </textarea></code>	definiert ein mehrzeiliges Eingabefeld. Sie müssen Höhe und Breite des Eingabebereichs angeben. Das Attribut <i>rows</i> = "...“ bestimmt die Anzahl der angezeigten Zeilen und damit die Höhe, während <i>cols</i> = "...“ die Anzahl der angezeigten Spalten festlegt. Spalten bedeutet dabei die Anzahl der Zeichen pro Zeile. Wenn Sie mit dem Composer arbeiten, vergessen Sie nicht, das schließende Tag <code></textarea></code> als eigenes Tag einzufügen.
<code><input type="hidden" name="FeldA1"></code>	definiert „versteckte“ Eingabefelder, wie sie von Applets genutzt werden, um die Eingaben als String an den WebAssign Server zu senden bzw. zu holen.
<code><input type="checkbox" name="FeldA1" value="Wert" ></code>	definiert „Kästchen zum Abhaken“, wie sie vom Aufgabenerstellungsassistenten eingesetzt werden, um Mehrfach-Auswahlen zu realisieren.
<code><input type="radio" name="FeldA1" value="Wert"></code>	definiert „Radiobox“, wie sie vom Aufgabenerstellungsassistenten eingesetzt werden, um Einfach-Auswahlen zu realisieren.

Abb. 28 : Die Typen von Eingabefeldern, die WebAssign kennt¹.

1. Weitere HTML- Select- Statements können mit Checkboxes und Radiobutton abgedeckt werden.

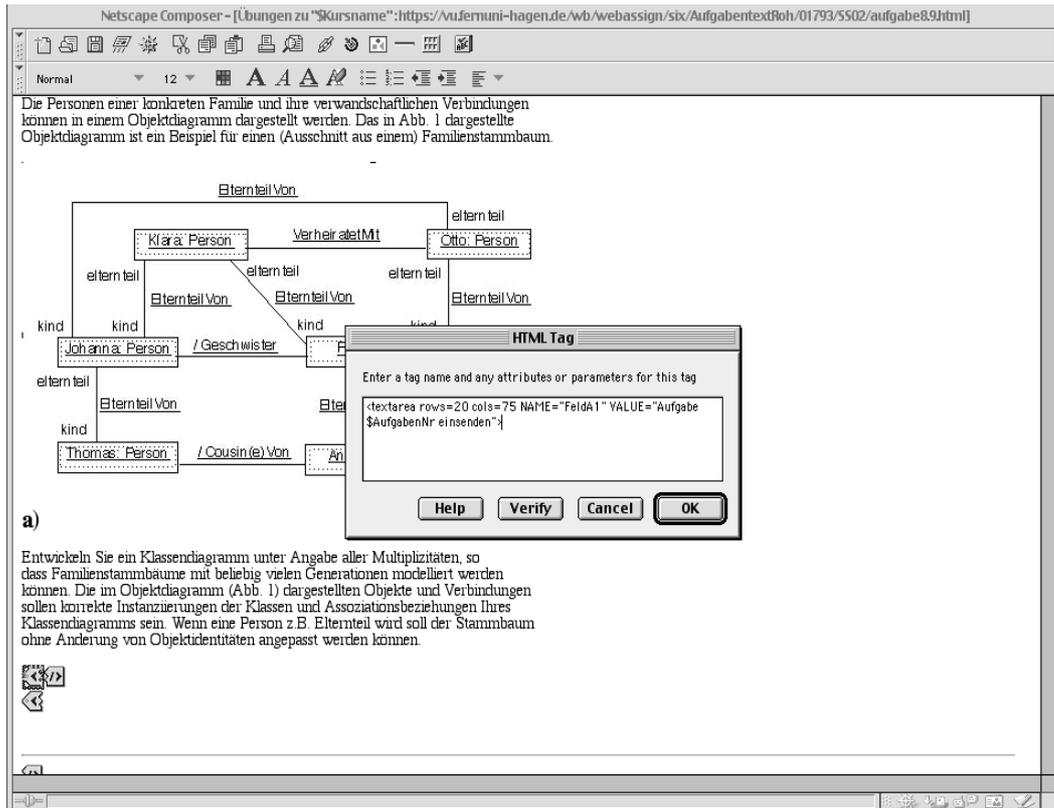


Abb. 29: Editieren der HTML-Aufgabenseite im Netscape Composer.

Benötigen Sie weitere Eingabefelder, z.B. für Teilaufgaben, erzeugen Sie diese am einfachsten, indem Sie das bereits bestehende Eingabefeld an die Stellen, wo Sie weitere Eingabefelder benötigen, kopieren. Anschließend benennen Sie im Eingabefeld-Tag das NAME-Attribut um (siehe Abb. 30). Die **Eingabefelder der ersten Teilaufgabe** werden mit *FeldA1*, *FeldA2* usw. und die **Eingabefelder der zweiten Teilaufgabe** mit *FeldB1*, *FeldB2* usw. bezeichnet, usf. **Alle** eingabefelder müssen stets dieser WebAssign Notation folgen. Wenn Sie weitere HTML-Tags benötigen, wählen Sie aus dem Browser Menü *Insert HTML TAG* aus.

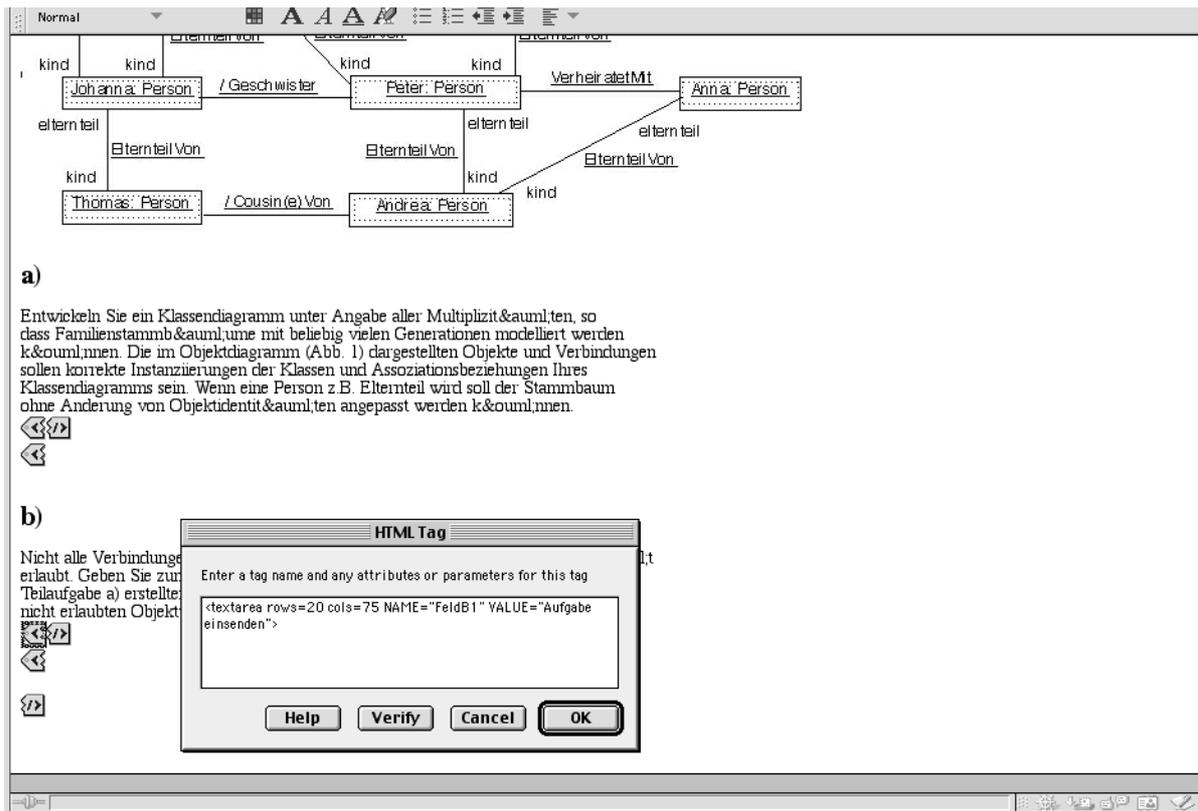


Abb. 30: (Um-) Benennung der Eingabefelder: Doppelklick auf gelbes Pfeilsymbol. Das erste Eingabefeld der zweiten Teilaufgabe heißt „FeldB1“.

Mit *Verify* können Sie die Syntax Ihrer Eingabe überprüfen¹, mit *Cancel* können Sie Ihre Änderungen „ungeschehen machen“ mit *OK* bestätigen Sie Ihre Änderung. Sie können, sofern Sie HTML beherrschen, direkt die HTML-Seite editieren, d.h. alle notwendigen Eingabefelder und Einsendebutton anlegen und den Aufgabentext angeben. Dabei müssen Sie die Konventionen zur Benennung der FORM-Elemente einhalten.

Grundsätzlich reicht ein FORM-Element für alle Teilaufgaben. Davon abweichend kann man auch für einzelne Teilaufgaben eigene FORM-Elemente definieren. Dies ist zum Beispiel erforderlich, wenn sich die Teilaufgabe in einem eigenen Frame befindet. Für jedes FORM-Element² gelten die gleichen Regeln wie für das FORM-Element, welches die ganze Aufgabe umfasst.

4.5.4 Anlegen weiterer Einsendebutton

Kopieren Sie den bereits angelegten Einsendebutton an die Stellen, an denen Sie weitere Einsendebutton benötigen, d.h. unter die Eingabefelder. Benennen Sie danach die Einsendebutton so um, dass sie *einsendenA*, *einsendenB*, usw. heißen. Um den Namen des Einsendebutton zu ändern, klicken Sie das gelbe Pfeilsymbol (analog zu Abb. 30) an und ändern Sie den Namen.

Ein Einsendebutton ist vom Typ *submit*. *Value* gibt an, was auf dem Button steht, den die Studie-

1. Beachten Sie, dass der Composer für jedes Tag, sowohl für das Öffnende z.B. `<textarea>` als auch für das Schließende z.B. `</textarea>` ein einzelnes Tag d.h. „einen gelben Pfeil“ benötigt.
2. Man beachte, dass Forms nicht geschachtelt werden können. Wir gehen hier davon aus, dass Sie über entsprechende HTML-Kenntnisse verfügen.

renden anklicken.

```
<INPUT TYPE="submit" NAME="einsendenA" VALUE="Aufgabe 1 einsenden">
```

Passen Sie auch das Attribut `VALUE` an, so dass statt *Aufgabe 1 einsenden* dort *Teilaufgabe <n> einsenden* steht. Mit *Verify* können Sie die Syntax Ihrer Eingabe überprüfen, mit *Cancel* können Sie Ihre Änderungen „ungeschehen machen“ mit *OK* bestätigen Sie Ihre Änderung. Sie können, sofern Sie HTML beherrschen, direkt die HTML-Seite editieren, d.h. alle notwendigen Eingabefelder und Einsendebutton anlegen und den Aufgabentext angeben. Dabei müssen Sie die Konventionen zur Benennung der HTML-Elemente einhalten.

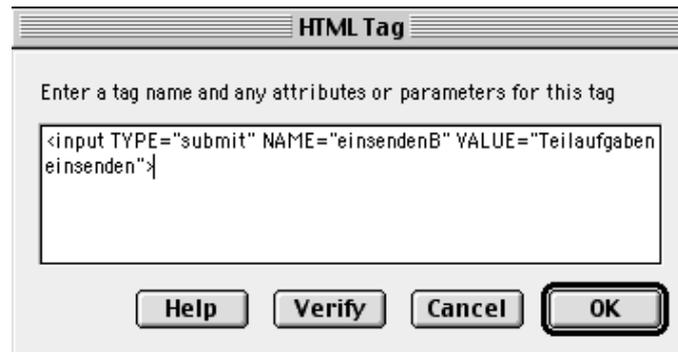


Abb. 31: Für die zweite Teilaufgabe werden die Attribute `NAME` und `VALUE` geändert.

Publizieren Sie den Aufgabentext und kontrollieren Sie das Erscheinungsbild (siehe Abschnitt 4.13).

4.6 Bewertungs- und Selbsttestmodule

Korrektur- und Selbsttestmodule sind beides „Programme“, die studentische Lösungen „verarbeiten“. Dabei unterscheiden sie sich wie folgt:

Ein **Bewertungsmodul** (ältere Bezeichnung: Korrekturmodul) korrigiert und bewertet die studentische Lösung und trägt die erreichte Punktzahl in die Tabelle der Ergebnisse für die Studierenden ein. (Folgen Sie dem Link *Ihre persönliche Ergebnisse* auf der *Studentenstartseite* (siehe *Handbuch für Studierende*.) Diese Punkte zählen beispielsweise bei einigen Kursen der FernUniversität Hagen zur Klausurzulassung.

Nutzen Sie ein Bewertungsmodul für automatisch korrigierbare Aufgaben, wenn die erreichte Punktzahl für die Studierenden und den Kursveranstalter relevant ist. Wenn Sie ein Bewertungsmodul zur automatischen Aus- und Bewertung spezifizieren, wählen Sie als Korrekturmodus *automatisch* aus. Das Bewertungsmodul wird z.B. vom *Aufgabenerstellungsassistenten* spezifiziert, wenn die Auswertung erst nach dem Bearbeitungsende erfolgen soll.

Ein **Selbsttestmodul** (ältere Bezeichnung: Vorkorrekturmodul) verarbeitet die studentische Lösung direkt bei der Einsendung und das Ergebnis wird in die Quittungsschablone eingebunden und so den Studierenden direkt zur Verfügung gestellt. Diese Verarbeitung kann eine Vorkorrektur eines Computerprogramms sein oder zukünftig eine Umwandlung vom Postscript-Format ins PDF-Format beinhalten. Wenn Sie nur Selbsttestmodule spezifizieren (z.B. bei automatisch auswertbaren Aufgaben als Trainingsmaterial) wählen Sie den Korrekturmodus *keine* aus. Es werden keine Punkte in die Tabelle der erreichten Punkte in „*Ihre persönliche Aufgabenübersicht*“ eingetragen.

Das Selbsttestmodul wird z.B. vom *Aufgabenerstellungsassistenten* spezifiziert, wenn die Aus-

wertung sofort nach dem Einsenden erfolgen soll. Mit der Quittung erhalten die Studierenden sowohl ihre eigenen Antworten als auch die korrekten Antworten zurück. Nutzen Sie diese Option, wenn die Ergebnisse der Studierenden für den Veranstalter nicht relevant sind, z.B. wenn Sie den Studierenden die Möglichkeit zum Üben geben möchten.

Wenn Sie, wie bei Programmieraufgaben oder künftig bei Aufgaben mit einer Konvertierung von Formaten, ein Selbsttestmodul angeben, die Aufgaben aber schließend manuell von einem Korrektor bewertet werden, geben Sie den Korrekturmodus *automatisch + von hand* aus. Bei diesen Aufgaben können Sie für jede technische Teilaufgabe ein eigenes Selbsttestmodul angeben.

In diesem Handbuch gehen wir davon aus, dass die Korrektur- und Selbsttestmodule zur Verfügung stehen und Sie als Autor nur das passende Bewertungsmodul angeben müssen. Fragen Sie in diesem Fall Ihre Betreuer.

4.7 Automatisch auswertbare Aufgaben ohne den Aufgabenerstellungs-Assistent

Wenn Sie das Spektrum der automatisch auswertbaren Aufgaben erweitern möchten, können Sie eigene Aufgabentypen erstellen. Hierzu benötigen Sie neben HTML-Kenntnissen auch Programmierkenntnisse, um das Programm zur Korrektur, das Korrektur- bzw. Selbsttestmodul, erstellen zu können. Beispiele für Bewertungs- und Selbsttestmodule sowie eine Anleitung finden Sie im *Handbuch für Betreuer*. An dieser Stelle gehen wir davon aus, dass die Module bereits existieren und zur Benutzung bereit stehen.

Abschnitt 4.6 erklärte, wie Sie Bewertungs- und Selbsttestmodule einsetzen können. Bevor es den Aufgabenerstellungsassistenten gab, nutzten bereits einige Veranstalter WebAssign für automatisch auswertbare Aufgaben. In diesen Fällen wurden Ankreuzfelder verwendet, d.h.: Checkboxen oder Radiobutton, um „1 aus n“ bzw. „n aus m“ Auswahlen zu realisieren. Dazu müssen folgende HTML-Konventionen zur Bezeichnung beachtet werden:

Bei Eingabefeldern vom Typ *checkbox* oder *radio* müssen alle Ankreuzmöglichkeiten, die zusammengehören¹, den gleichen Namen haben und sich in den VALUE-Attributen unterscheiden. Die VALUE-Attribute können Sie frei wählen für Multiple-Choice-Aufgaben haben sich in der FernUniversität beispielsweise *A, B, C* usw. eingebürgert. Beispiel:

```
<INPUT TYPE="checkbox" NAME="FeldA1" VALUE="A">  
...  
<INPUT TYPE="checkbox" NAME="FeldA1" VALUE="E">
```

Für die Gestaltung der Aufgaben stehen Ihnen die Möglichkeiten von HTML zur Verfügung (siehe auch Abschnitt 4.9 *Fortgeschrittene Gestaltung von Aufgabentexten*).

4.8 Aufgaben mit Pre-Testing

Aufgaben mit einem automatischen Pre-Testing und einer anschließenden manuellen Korrektur sind vom Korrekturmodus *automatisch + von hand*. Aufgaben zu Selbsttestzwecken sind vom Korrekturmodus *keine*. Sie erstellen sie analog zu Freitext-Aufgaben. Dabei können Sie z.B. ein Eingabefeld und einen Einsendebutton für das Programm oder mehrere Eingabefelder² und Ein-

-
1. Die zusammengehörigen *checkbox*- oder *radio*-Inputs werden als **ein** Eingabefeld aufgefasst.
 2. Beachten Sie, dass der Name der Eingabefelder für Programmcode in der Korrektorschablone und den Quittungen von der Form *FeldAP* sein muss (siehe Abschnitt 4.14 und 4.15).

sendebuttone für mehrere Programmstücke anlegen. Lassen Sie die Teilaufgaben erzeugen und weisen Sie den Teilaufgaben die entsprechenden Selbsttestmodule zu. Den Namen des Selbsttestmoduls müssen Sie per Hand angeben, wenden Sie sich hierzu an Ihre Betreuer, falls Sie die Bezeichnung des Vorkorrektormoduls nicht kennen. Wie Sie eigene Selbsttestmodule schreiben, ist im *Handbuch für Betreuer* angegeben.

4.9 Fortgeschrittene Gestaltung des Aufgabentexts

Bei der Erstellung der HTML-Seite müssen Sie nur wenige Regeln einhalten. Diese betreffen zum einen den Kopf des HTML-Dokuments, d.h. alles was zwischen den Head-Tags <HEAD>...</HEAD/> steht und zum anderen die Bezeichnung von Eingabefeldern und Button.

Im Dokumentenkopf, genauer in den Angaben zum FORM-Tag, wird festgelegt, um welche Aufgabe es sich handelt und welches Server-Programm die Einsendung entgegen nimmt. Durch den Einsatz von Variablen kann der Dokumentenkopf für alle Aufgaben gleich gehalten werden, beispielsweise:

```
<HTML>
  <HEAD>
    <TITLE>Übungen zu "$Kursname"</TITLE>
  </HEAD>
  <BODY>
    <FORM
      METHOD="POST"
      ACTION="$WebAssignServer/$Veranstaltername/Einsendung/$KursNr/$VersionsNr/$AufgabenheftNr/$AufgabenNr"
      NAME="Aufgabe">

    <H2>Übungen zu "$Kursname"</H2>
    <P><FONT SIZE="+1">$Versionsname</FONT></P>
    <P><FONT SIZE="+1">Aufgabenheft $AufgabenheftNr</FONT></P>
    ...
```

Sie können von diesem Muster abweichen, solange die Angaben im FORM-Tag stimmen. Um ein gleiches Erscheinungsbild für alle Aufgaben eines Kurses zu gewährleisten, sprechen Sie Änderungen am FORM-Tag und an den Variablen auf jeden Fall zuvor mit den Kursbetreuern ab. Lesen Sie auch das *Handbuch für Betreuer*, bevor Sie Variablen ändern.

Publizieren Sie stets den Aufgabentext und weitere von Ihnen geänderte Seiten und kontrollieren Sie das Erscheinungsbild (siehe Abschnitt 5).

4.9.1 Verwendung von Grafiken und Applets in Aufgaben

Zur Gestaltung der Aufgabentexte können Sie alle Möglichkeiten von HTML ausnutzen, einschließlich Grafiken, Tabellen, Frames usw. Wenn Sie in einen Aufgabentext Frames und Tabellen etc. verwenden möchten, sollten Sie HTML sicher beherrschen. Die Applets und Grafiken, d.h. die GIF-, JPEG-Dateien werden als Kursressourcen¹ bezeichnet. Die Ressourcen müssen im selben Verzeichnis (Directory) liegen, wie die HTML-Seite, in die sie eingebunden sind und Sie müssen alle Ressourcen über lokale Referenzen in die Aufgaben einbinden.

Beispiel:

- Wenn Sie die Grafik *bild.jpg* in die Aufgabenseite einbinden wollen, lautet das HTML-TAG: . Beachten Sie, dass Sie Bilder nicht über den Menüpunkt *Insert Image* des Composers einbinden können.

1. Weiteres über Kursressourcen lesen Sie im Handbuch für Benutzer.

- Eine Datei, die den Aufgabentext enthält z.B. eine unvollständige Grafik, die die Studierenden vervollständigen sollen, binden Sie über einen Link ein:

`datei`

In diesem Fall sollten Sie aber auf die Größe der Datei, d.h. die Download-Zeiten achten.

Geben Sie das Verzeichnis der Aufgabenseite und ihrer Ressourcen beim Publizieren als *include files* an. Beachten Sie weiter, dass die Namen der Ressourcen innerhalb des Kurses bzw. der Kursversion eindeutig sein müssen und das Groß- und Kleinschreibung unterschieden wird.

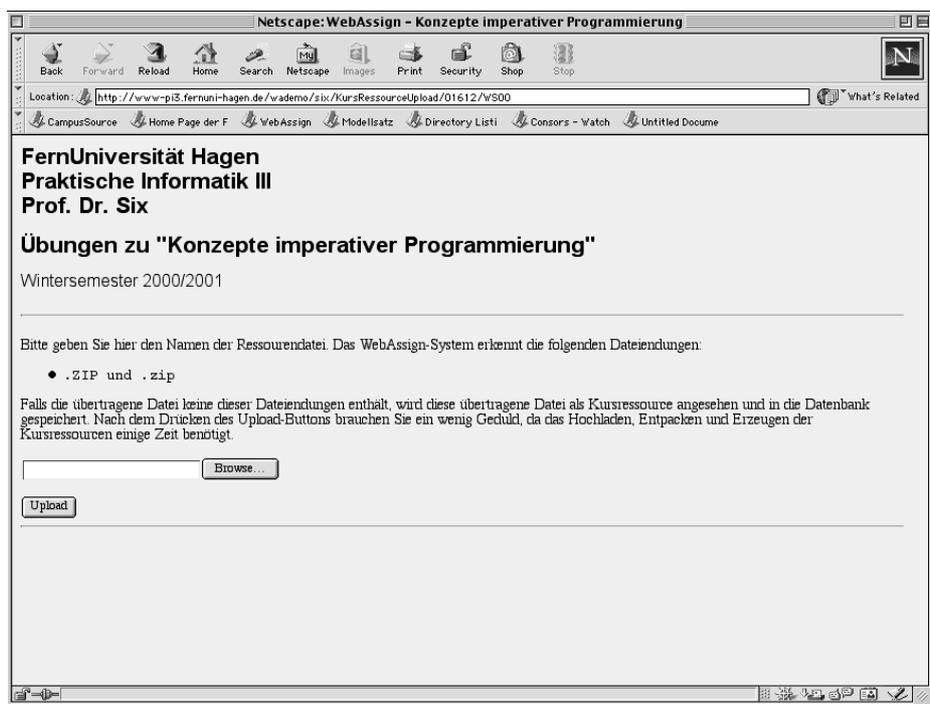


Abb. 32: Kursressourcen-Upload

Bilder, Grafiken, Applets oder auch Aufgabentexte als Dateien beliebiger Formate werden als zusätzliche Kursressourcen bezeichnet. Diese müssen Sie auf den WebAssign-Server hochladen. Wählen Sie dazu den Link *KursRessourcen-Upload* auf der Benutzerstartseite (siehe Abb. 32) in der *Aufgabenverwaltung* aus.

Sie können den Dateinamen direkt in das Eingabefeld eingeben oder über den Button *Durchsuchen...*- bzw. *Browse...* die Datei auswählen. Laden Sie die ausgewählte Datei anschließend mit dem *Upload* Button auf den Server hoch.

Derzeit können zip-Dateien sowie ungepackte Einzeldateien hochgeladen werden. Wenn Sie mit einem PC arbeiten, sollten Sie die Dateien **immer** zuvor zippen, da unter Windows einzelne Dateien zu Problemen mit den Pfadnamen führen.

4.9.2 Verwendung von Grafiken in Quittungsseiten

Eine Besonderheit ist für zusätzliche Kursressourcen, d.h. Bilder und Applets in Quittungsseiten, zu beachten:

In Quittungen können Sie, im Gegensatz zu Aufgabenseiten, keine relativen Links wie `` verwenden, sondern Sie müssen die absoluten Adressen wie

 angeben.

4.10 Aufgaben, zu denen Studierenden mit beliebigen Editoren die Lösung erstellen

Wenn Ihnen die Möglichkeiten von HTML nicht ausreichen und Sie den Studierenden die Möglichkeit geben möchten, die Lösungen mit einem vorgegebenen Editor oder Programm zu erstellen (z.B. mit einem speziellen Textverarbeitungs- oder Bildbearbeitungsprogramm), wählen Sie diese Option, bei der die Studierenden ihre Lösungen als Dateien eines beliebigen Typs auf den WebAssign-Server hochladen. Beachten Sie bitte, dass diese Dateien recht groß werden können und dass langsame Modemverbindungen das Einsenden der Lösung zu einem Geduldsspiel werden lassen können. Es empfiehlt sich zudem, allen Studierenden und Korrektoren denselben Editor bzw. das gleiche Programm vorzuschreiben und den Studierenden und Korrektoren mitzuteilen, wie sie ihren Browser für den Dateityp konfigurieren, so dass die Dateien vom Browser mit dem entsprechenden Programm direkt geöffnet werden.

Im Folgenden wird beschrieben, wie Sie entsprechende Aufgabenseiten in WebAssign erstellen.

Wählen Sie auf der Aufgabenerstellungsseite in der Combobox *Studierendenlösung als externe Datei* Ja aus und lassen Sie das Grundgerüst einer Aufgabe, bei der die Studierenden beliebige Dateien auf den WebAssign-Server hochladen können, mit dem Button *generieren* generieren. Damit haben Sie eine Freitext-Aufgabe mit einem Eingabefeld vom Typ FILE angelegt. Das Form-Tag sieht folgendermaßen aus (siehe Abb. 34).

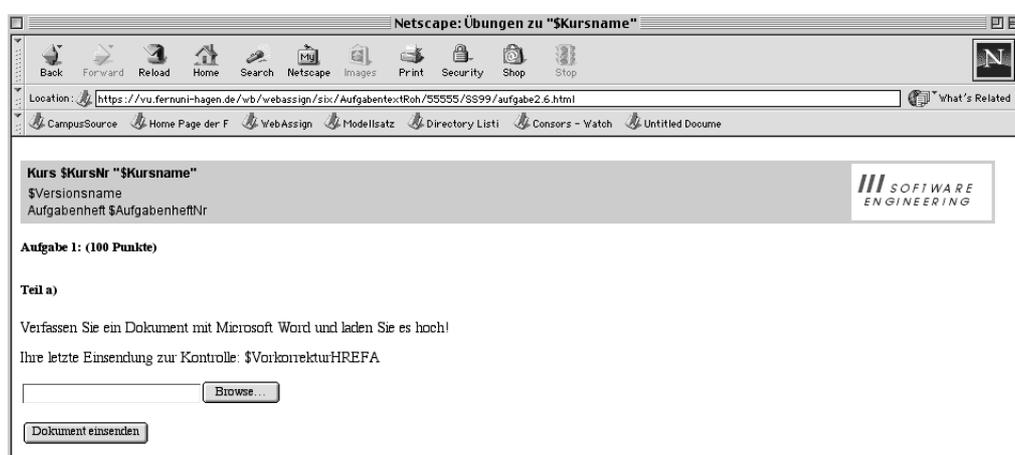


Abb. 33: Die generierte Standardaufgabe mit einem HTML-Eingabefeld wurde ohne Änderungen publiziert. Den Variablen werden später Werte zugewiesen.

```
<FORM
  METHOD="POST"
  ENCTYPE="multipart/form-data" ACTION="$WebAssignServer/$Veranstaltername/
  EinsendungMultipart/$KursNr/$VersionsNr/$AufgabenheftNr/$AufgabenNr">
  ...
  <INPUT TYPE="file" name="FeldA1">
  <INPUT TYPE="submit" NAME="einsendenA" VALUE="Datei einsenden">
</FORM>
```

Abb. 34: FORM-Tag für Aufgaben mit Datei-Upload¹.

4.10.1 Einbindung der studentischen Lösung in die Aufgabe zur erneuten Bearbeitung

WebAssign Standard ist, dass den Studierenden, wenn Sie eine Aufgabe erneut bearbeiten, ihre vorherige Lösung angezeigt wird. Im Falle von externen Dateien ist das Standard-Verhalten von WebAssign anders. Die vorhergehende Lösung wird nicht automatisch angezeigt, da davon ausgegangen wird, dass die Studierenden ihre Dateien lokal gespeichert haben.

Wenn Sie dagegen die bisherige Lösung den Studierenden zur Verfügung stellen möchten, binden Sie die Variable `$FeldA1` in der gewünschten Form in die HTML-Seite der Aufgabe ein (siehe Abb. 35). Dabei können Sie über die Variable `$FeldA1` die Behandlung bzw. Darstellung der Datei auf den Rechner der Studierenden beeinflussen (siehe Abb. 36).

```
<FORM ...>
  Aufgabe 3:
  (Fragestellung)
  Ihre letzte Einsendung zur Kontrolle: $FeldA1HREF
  <INPUT ... TYPE="FILE" ...>
  <INPUT ... TYPE="SUBMIT" ...>
</FORM>
```

Abb. 35: Die Variable `$FeldA1HREF` bindet bei erneuten Bearbeitungen die bisherige studentische Lösung über einen Link in die Aufgabe ein.

1. Falls der VALUE „Datei einsenden“ nicht im INPUT -TAG vom Typ submit steht, sondern z.B. Aufgabe <n.n> einsenden, ersetzen Sie dies durch Datei einsenden.

Wählen Sie

\$FeldA1	so entscheidet der Browser des Studierenden je nach MIME-Typ, wie die Datei dargestellt wird:	
	Mimetyyp	Darstellungsform
	image/jpeg image/gif image/png	IMG,d.h. als Bild
	text/* (außer text/ps)	PLAIN,d.h. als Klartext, sofern es kein Postscript ist.
	sonst	HREF, d.h. als Link
\$Feld A1URL	so wird der Dateipfad der Datei, angezeigt	
\$Feld A1HREF	so wird die Datei in ein Link-Tag eingebettet	
\$FeldA1IMG	so wird die Datei in ein Image-Tag eingebettet	

Abb. 36: Einbindung der studentischen Lösung.

4.10.2 Korrekturschablonen

Wählen Sie neben dem Eintrag *Korrekturseite* den Button *Schablone generieren* aus, um eine Korrekturschablone zu generieren. Die so generierte Standardschablone bindet die studentische Lösung mit den Variablen \$FeldA1BK über einen Link in die Korrekturseite ein, d.h. die Korrektoren sehen einen Link auf die Datei mit der studentischen Lösung. Sie können über die Variable $\$FELDA1BK\{URL/HREF/IMG/PLAIN\}$ die Darstellung beeinflussen. Die Variablen verhalten sich analog zu den Variablen in der Aufgabenseite, d.h. zur Beschreibung in Abb. 36.

4.10.3 Quittungen

Wählen Sie neben dem Eintrag *Quittungsseite* den Button *Schablone generieren* aus .

4.11 Aufgaben, die von Autoren mit beliebigen Editoren erstellt werden

In bestimmten Fällen kann es sinnvoll sein, die Aufgabentexte nicht als reine HTML-Seite anzulegen, z.B. wenn die Studierenden „den Aufgabentext bearbeiten“ sollen und dazu ein spezielles Programm (CAD-, Grafik-, Tabellenverarbeitungsprogramm) benötigt wird.

Als Autor erstellen Sie dann die Aufgabe mit dem Programm und binden sie über einen Link in die HTML-Aufgabenseite ein (siehe Abschnitt 4.9).

Sie können den Link selbst anlegen oder wie im Abschnitt 4.3 beschrieben vorgehen. Beachten

Sie aber bitte, dass sich durch externe Dateien die Downloadzeiten und -kosten für die Studierenden erhöhen.

4.12 Konsistenzerhaltung nach Änderungen einer Aufgabe

Nach einer Änderung der Aufgabenseite müssen Sie deren syntaktische Korrektheit erneut über den Button *Vorlage analysieren* überprüfen lassen. Hat sich die Anzahl der technischen Teilaufgaben geändert, müssen Sie über den Button *Teilaufgaben erzeugen* WebAssign anweisen, diese Informationen zu verarbeiten und in die Datenbank zu übernehmen.

Weiterhin müssen Sie die Konsistenz der Quittungen und der Korrekturschablone von Hand überprüfen. Sie erzeugen die Quittungs- und Korrekturseiten¹ über die entsprechenden *generieren*-Button neu.

4.13 Publizieren einer Aufgabe

Nachdem Sie die Aufgabe fertiggestellt haben, müssen Sie sie auf dem WebAssign-Server publizieren. Nutzen Sie dazu die Publish Funktion des Netscape Browsers (siehe Abb. 38), die Sie im Browser-Menü unter *Datei-Seite-Publizieren* finden. Editieren Sie den Dateinamen im Feld *Publish files to this location (FTP or HTTP)*: indem Sie an den dort angegebenen Namen wie folgt ergänzen; die anderen Felder können frei gelassen werden und falls dabei eine Fehlermeldung auftaucht, können Sie diese ignorieren.

Aufgabenseiten	aufgabe <n.n>.html
Korrekturschablone	korrektur <n.n>.html
Quittungsseiten	quittungen <n.n.n>.html
Musterlösungsseite	muster <n.n>.html

Abb. 37: Ergänzung des Pfadnamens beim Publizieren.

1. Quittungsseiten von nicht mehr vorhandenen Teilaufgaben werden von WebAssign automatisch gelöscht.

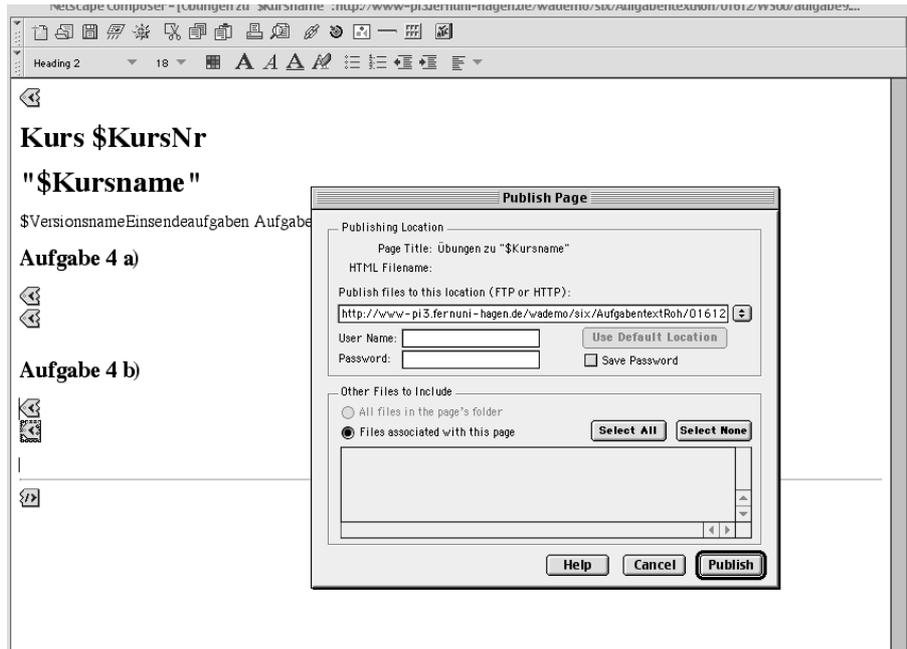


Abb. 38: Publizieren einer Aufgabe.

4.14 Korrekturseite

Für jede Aufgabe ist eine Korrekturseite obligatorisch. Sie erzeugen sie, indem Sie den Button *Korrekturseite: korrektur <n:m>.html generieren* auf der Aufgabenerstellungsseite wählen. Standardgemäß werden Informationen zur Aufgabe, zum Studierenden und die studentische Lösung über Variablen eingebunden (siehe Abb. 39). Die Information über die Aufgabe beinhalten den Kursnamen, die Nummer des Aufgabenheftes, den Aufgabennamen, die Informationen über die Studierenden sind Vorname und Name.

Sinnvolle Ergänzungen sind die Matrikelnummer und der Name des Korrektors. Diese werden von den Kursbetreuern vorgegeben und sollten nicht geändert werden, um ein einheitliches Er-

scheinungsbild zu gewährleisten.

Netscape: Einsendung zu "\$Kursname", Aufgabenheft \$AufgabenheftNr, \$Aufgabenname

Location: <https://vu.fernuni-hagen.de/vb/webassign/stx/AufgabentextRoh/01612/WS01/korrektur2.1.html>

Einsendung zu "\$Kursname", Aufgabenheft \$AufgabenheftNr, \$Aufgabenname
von \$Vorname \$Nachname
Korrektor: \$Korrektor
Datum: {hier bitte das Datum der Korrektur eintragen}

Compilerausgabe Testläufe Prettyprinter Bewertung

Eingabe in Feld 1:
\$FeldA1
\$VorkorrekturA

Anmerkungen
{hier bitte zusätzl. Anmerkungen eintragen}
Erreichte Punktzahl: {hier bitte die erreichte Punktzahl eintragen}

Abb. 39: Korrekturschablone.

Die Variable *\$Vorkorrektur* bindet bei Aufgaben mit automatischer Vorkorrektur, z.B. bei Programmieraufgaben mit Pre-Testing, die Ergebnisse der Vorkorrektur ein. In allen anderen Fällen wird diese Variable ignoriert.

Erwarten Sie in den Eingabefeldern Programme oder Teile von Programmen, so müssen Sie ein *P* an den Name des Eingabefeldes in Quittungs- und Korrekturseiten anfügen, z.B. *FeldA1P*. Das *P* sorgt dann dafür, dass in der Korrekturseite beim Korrektor z.B. das „kleiner-Zeichen“ *<* in Programmen nicht als Beginn eines HTML-Tags interpretiert wird (siehe Abb. 41). Wir raten davon ab, „vorsorglich“ stets ein *P* anzuhängen, da damit in Einsendungen und Vorkorrekturen vorkommende und richtige HTML-Tags dann ebenfalls zu *<* bzw. *>* der HTML-Codierung des *<* bzw. *>* Zeichens expandiert werden.

Netscape: Einsendung zu "\$Kursname", Aufgabenheft \$AufgabenheftNr, \$Aufgabenname

Location: <https://vu.fernuni-hagen.de/vb/webassign/stx/AufgabentextRoh/01612/WS01/korrektur2.1.html>

Einsendung zu "\$Kursname", Aufgabenheft \$AufgabenheftNr, \$Aufgabenname
von \$Vorname \$Nachname
Korrektor: \$Korrektor
Datum: {hier bitte das Datum der Korrektur eintragen}

Compilerausgabe Testläufe Prettyprinter Bewertung

Eingabe in Feld 1:
\$FeldA1P
\$VorkorrekturA

Eingabe in Feld 2:
\$FeldA2P
\$VorkorrekturA

Anmerkungen
{hier bitte zusätzl. Anmerkungen eintragen}
Erreichte Punktzahl: {hier bitte die erreichte Punktzahl eintragen}

Abb. 40: Korrekturseite für Programmieraufgaben : *\$FeldA1P*.

Hinweise, die die Korrektoren bei der Bewertung der studentischen Lösung beachten sollen,

können Sie als Korrekturhinweise notieren: Wählen Sie hierzu *Korrekturhinweis: hinweise <n.m>* aus. Wenn Sie die Korrekturseite geändert haben, müssen Sie sie anschließend publizieren. Beachten Sie, dass Sie beim Publizieren ggf. *korrektur<n.n>.html* anfügen müssen.

4.15 Quittungen

Nachdem die Studierenden ihre Lösungen eingesendet haben, erwarten Sie in der Regel eine Bestätigung oder Quittung, dass ihre Lösungen den WebAssign-Server erreicht haben. Die generierte Standard-Quittung reicht in den meisten Fällen aus. Dabei werden die Eingaben wiederholt sowie Angaben zur Aufgabe und der Name des Studierenden ausgegeben.

Für jede (technische) Teilaufgabe, d.h. für jeden Einsendebutton einer Aufgabe benötigen Sie eine Quittung. Nachdem Sie für die Teilaufgaben mit dem Button *Teilaufgaben generieren* die entsprechenden Eingabefelder in der Datenbank angelegt haben, zeigt WebAssign auf der *Aufgabenerstellungsseite* im Teilaufgabenbereich die technischen Teilaufgaben an. Für diese müssen Sie mit dem Button *Teil <n> quittung <n.n.x>.html generieren* jeweils eine Quittung generieren.

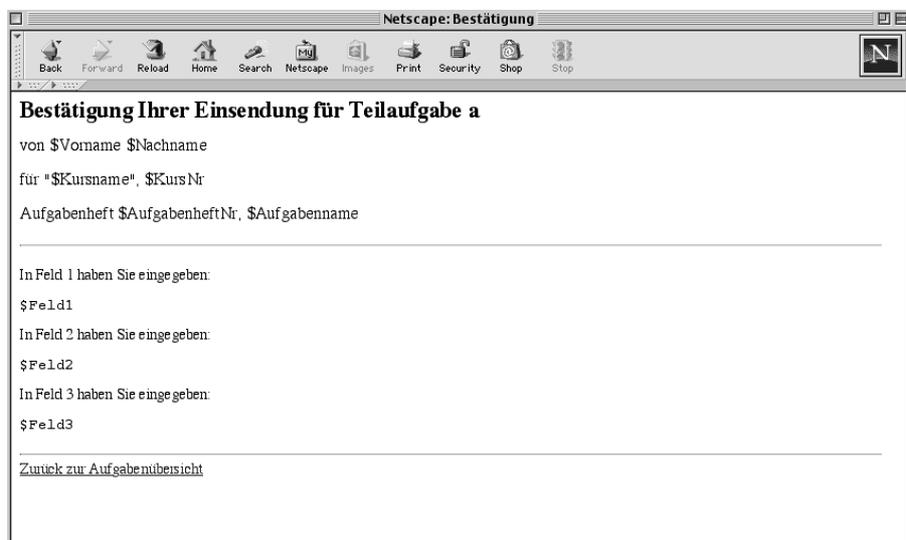


Abb. 41: Quittung einer Aufgabe mit 3 Eingabefeldern.

Erwarten Sie in den Eingabefeldern Programme oder Teile von Programmen, müssen Sie ein *P* an den Namen des Eingabefeldes in der Quittungsseite anfügen, z.B. Feld1P. Das *P* sorgt dann dafür, dass in der Quittung z.B. das „kleiner-Zeichen“ < in Programmen nicht als Beginn eines

HTML-Tags interpretiert wird.

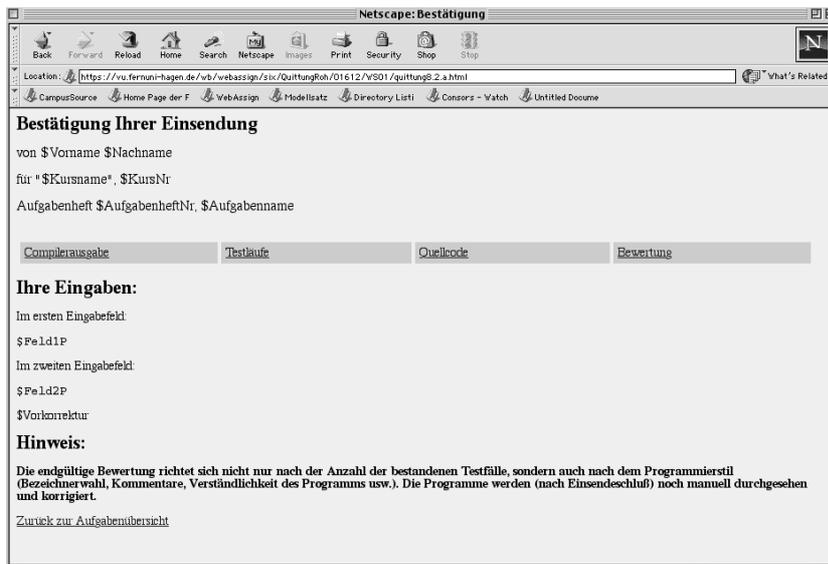


Abb. 42: Quittungsseite für Programmieraufgaben : \$Feld1P

Wenn Sie die Quittungsseite geändert haben, müssen Sie sie publizieren. Beachten Sie dabei, dass Sie *quittung<n.n.n>.html* an den Dateinamen beim Publizieren anfügen müssen (siehe 4.13).

4.16 Musterlösung

Um den Studierenden, die Schwierigkeiten bei der Bearbeitung der Aufgaben hatten, das Verständnis des Stoffes zu erleichtern, sind in WebAssign Musterlösungen vorgesehen. Sie erzeugen die Musterlösungsseite, indem Sie *Musterlösung muster <n.m>.html* Schablone generieren auf der Aufgabenerstellungsseite auswählen. Es wird eine HTML Seite erzeugt (siehe Abb. 43). Öffnen Sie diese Seite im Composer und schreiben Sie die Musterlösung auf. Sichern und publizieren Sie die Musterlösung. Beachten Sie, dass Sie an den Dateinamen *muster<n.n>.html* anhängen müssen (siehe 4.13). Die Musterlösung wird den Studierenden gleichzeitig mit den korrigierten und bewerteten Lösung zur Verfügung gestellt.



Abb. 43: Musterlösung: Editieren Sie die Seiten im Composer und publizieren Sie sie.

5 Aufgaben überprüfen und testen

WebAssign bietet ab dem Release 1.14 einen automatischen Vollständigkeits- und Konsistenztest der Aufgaben eines Aufgabenhefts an. Wählen Sie zur Durchführung des Tests den Link

Vollständigkeitsüberprüfung auf der Betreuerstartseite aus (siehe Abb. 2). WebAssign überprüft die Existenz aller notwendigen Aufgabenseiten und Schablonen. Alle fehlenden Ressourcen werden aufgelistet und ein Link führt direkt zur entsprechenden Erstellungsmöglichkeit.

The screenshot shows a web browser window with the title 'Automatischer Aufgabencheck'. Below the title is a table with two columns: 'Fehlende Ressource' and 'Link zur Erstellung'. The table lists 18 rows of missing resources, each with a corresponding 'erstellen' link.

Fehlende Ressource	Link zur Erstellung
korrektur1.6.html	erstellen
quittung1.6.a.html	erstellen
muster1.6.html	erstellen
korrektur1.6.html	erstellen
quittung1.6.b.html	erstellen
muster1.6.html	erstellen
korrektur1.6.html	erstellen
quittung1.6.c.html	erstellen
muster1.6.html	erstellen
korrektur1.7.html	erstellen
quittung1.7.a.html	erstellen
muster1.7.html	erstellen
korrektur1.8.html	erstellen
quittung1.8.a.html	erstellen
muster1.8.html	erstellen
korrektur1.8.html	erstellen
quittung1.8.b.html	erstellen

Abb. 44 Ergebnis der automatischen Vollständigkeits- und Konsistenzüberprüfung

Zusätzlich testet WebAssign noch einige Konsistenzkriterien, z.B. ob bei Pre-Testaufgaben die Quittungsschablone die Variable \$Vorkorrektur enthält. Ist dies nicht der Fall, so könnte der Studierende keinen Einblick in das Ergebnis des Selbsttestmoduls bekommen.

Neben diesem automatischen Check sollten Sie als Autor auf jeden Fall das Layout der Aufgaben und ihre „funktionsweise“ überprüfen, d.h. Sie sollten sich die Aufgabe aus Sicht der Studierenden ansehen, sie als Student bearbeiten und als Korrektor korrigieren.

Wenn Sie die von Ihnen zu erstellenden Aufgaben ohne die „neugierigen Blicke“ der Öffentlichkeit, d.h. der Studierenden einstellen und testen wollen, wählen Sie für die betroffenen Aufgabenhefte einen Bearbeitungszeitraum, der in der Zukunft liegt. Somit sind die Aufgaben für reguläre Studierende nicht erreichbar. Nur der Teststudent 7777777, den einzurichten Sie den Betreuer bitten sollten, hat unabhängig vom Bearbeitungszeitraum Zugriff auf die Aufgaben.

Nachdem Sie die Aufgaben erstellt haben, werden Sie zum Studierenden 7777777, indem Sie auf der Aufgabenseite den Link *Studentenstartseite auswählen* und sich damit erneut als Studierender identifizieren.

Überprüfen Sie das Layout der Aufgabe, korrigieren Sie gegebenenfalls die Aufgabe nachdem Sie sich wieder als Autor angemeldet haben und wiederholen Sie den Prozess des Rollenwechsels erneut.

Schicken Sie schließlich als Student 7777777 eine Lösung ein (siehe *Handbuch für Studierende*). Überprüfen und korrigieren Sie gegebenenfalls die Quittung.

Haben Sie eine Korrektur *von hand* vorgesehen, schließen Sie als Student das Aufgabenheft. Jetzt bitten Sie den Betreuer Ihnen einen Account als Korrektor einzurichten und überprüfen Sie die Korrektorschablone. An dieser Stelle bietet es sich an, dass Sie als Autor auch ausgewählte Funktionen eines Betreuers wahrnehmen (siehe *Handbuch für Betreuer*), als Betreuer Hefteinsendungen der Studierenden rückgängig machen und Aufgaben erneut an die Korrektoren aufteilen.

Wiederholen Sie den Prozess so lange, bis die Aufgaben erwartungsgemäß von den Studierenden bearbeitet und von den Korrektoren korrigiert werden können.