Funktionen einer Übungskomponente für netzgestützte Kurse

Stephan Trahasch
Institut für Informatik
Albert-Ludwigs-Universität Freiburg
Georges-Köhler-Allee
79110 Freiburg
trahasch@informatik.uni-freiburg.de

12.07.2001

Einleitung

Mit zunehmender Tendenz werden an Hochschulen netzgestützte Distanz-Kurse als Ergänzungen des Curriculums angeboten. Die Forschung und die Entwicklung von Werkzeugen konzentrierte sich bisher auf die Phase der netzgestützten Inhaltspräsentation und -vermittlung. Um einen Kurs vollständig über das Internet anbieten zu können, muss auch die Übungsphase netzgestützt abgewickelt werden können. Aufgaben zur Selbst-Einschätzung der Studierenden, die von der Lehr-/Lernplattform automatisch ausgewertet werden können und den Studierenden ein Feedback geben, werden von den meisten Lehr-/Lernplattformen unterstützt. Der klassische Übungsbetrieb in den naturwissenschaftlich orientierten Fachgebieten ist hingegen bisher in keiner Lehr-/Lernplattform integriert. Unter dem klassischen Übungsbetrieb verstehen wir, dass während des Kurses Übungsaufgaben von dem Betreuer erstellt und publiziert werden, die dann von den Studierenden bearbeitet werden. Die Studierenden geben ihre Lösungen zu einem festen Termin ab; diese werden anschließend von Tutoren korrigiert und bewertet. Der Betreuer oder der Dozent besprechen nach dem Abgabetermin die Aufgaben mit den Studierenden.
Die Virtualisierung des Übungsbetriebes ist notwendig, damit alle Phase des Lehrens und Lernens über das Netz abgewickelt und somit Distanzkurse angeboten werden können.

Rollen

Alle Teilnehmer eines Kurses nehmen automatisch an den Übungen teil. Die Teilnehmer können folgenden Rollen zugeordnet werden, wobei eine Person verschiedene Rollen innehaben kann. Die Rollenzugehörigkeit gilt für den gesamten Kurs.

- Dozent
- Betreuer
- Tutor
- Studierender

Konzeption der Übungsaufgaben

Mehrere Aufgaben werden zu einem Aufgabenheft zusammengefasst. Eine Aufgabe besteht aus vier Elementen:

1. Aufgabe
2. Quittung/Eingangsbestätigung
3. Musterlösung (optional)
4. Korrekturhinweise (optional)

Die Aufgaben sind (X)HTML- oder XML-Dateien, die alle internetfähige Dokumentenarten enthalten können. Ein Import von Aufgaben, die dem IMS-Question & Test Interoperability Standard entsprechen, sollte mittelfristig möglich sein.
Aufgaben und Musterlösungen können von allen Anwendern aufgabenheftweise ausgedruckt werden. Es werden nur die Aufgabenstellung und die Musterlösung ohne zusätzliche Elemente der Lehr-/Lernplattform wie z.B. Navigations- oder Funktions-Buttons ausgedruckt.

Aufgabentypen

- Alle Aufgabentypen zum Self-Assessment können auch in bewerteten Aufgaben verwendet werden.
- Dateieinsendeaufgaben für verschiedene Formate (Gif, JPEG, Office-Dokumente etc.)
- Erweiterungen der Eingabemöglichkeiten durch Applets oder Flash (Schnittstelle offen legen, damit Fachbereiche eigene Erweiterungen entwickeln können)

Fachspezifische Anforderungen:
Informatik:

- Eingabe von mathematischen Formeln und Beweisen (MathML, Latex)
- Konzeption und Eingabemöglichkeit von Bäumen und Graphen
- Kompilieren und Auswertung von Programmen (Vorkorrektur)

Betreuer

Der Betreuer organisiert die Übungen, verwaltet die Teilnehmer und Teilnehmerinnen, konzipiert und entwickelt Übungsaufgaben.
Die Aufgaben können offline von technisch versierten Betreuern entwickelt und anschließend in das System eingefügt werden. Technisch nicht so versierte Betreuer werden durch einen „Online-Wizard“ bei der Aufgabenerstellung unterstützt.

Ein Aufgabenheft wird durch folgende Angaben parametrisiert:

- Bearbeitungsbeginn des Aufgabenhefts (Datum, Zeit)
- Einsendeschluss des Aufgabenhefts (Datum, Zeit)
- Korrektur nach Einsendeschluss oder nach Eingang der Aufgabeneinsendung
- Referenz zum Lernmodul (optional)
- Für alle Aufgabenhefte kann eine IP-Maske angegeben werden. Nur von Rechnern mit IP-Nummern, die der IP-Maske entsprechen, können die Aufgaben bearbeiten.

Eine Aufgabe (Aufgabe, Quittung, Musterlösung, Korrekturhinweise) wird durch folgende Angaben parametrisiert:
- Maximale Punktzahl
- Veröffentlichungsdatum der Musterlösung (frei definierbar und nach Einsendeschluss; Standard: nach Einsendeschluss)
- Maximale Einsendemöglichkeiten (1 bis unendlich; Standard: unendlich)
- Zeitliche Begrenzung der Aufgabenbearbeitung (Standard: keine Begrenzung)
- Referenz zum Lernmodul (optional)

Jede Aufgabe (Aufgabe, Quittung, Musterlösung, Korrekturhinweise) wird in einem Aufgabenpool aufgenommen und durch weitere Metadaten beschrieben (Sprache, Schwierigkeitsgrad, Stichworte, Fachbereich, Kurs, Rechte etc.)

Über Reports (Ergebnisse, erreichte Punktzahl der Studierenden, Verlauf der Einsendungen und Korrekturen, frei definierbare Reports) und Statistiken (durchschnittliche Punktzahl, Anzahl der Einsendungen etc.) kann der Betreuer den Verlauf der Übungen analysieren. Reports und Statistiken können exportiert werden, damit diese in anderen Programmen weiterverwendet werden können (z.B. Scheine drucken).
Falls im Kurs eine Klausur geschrieben wird, so kann der Betreuer diese über das System ankündigen und die Ergebnisse in die persönliche „Aufgabenmappe“ der Studierenden eintragen.

Weitere Funktionalitäten für Betreuer:
- Anmelden und Verwaltung von Tutoren und Dozenten. Dies erfolgt bei der Erstellung des Kurses.
- Freigabe von Korrekturen
- Zugang zu allen Korrekturen und Einsendungen
- Verändern der Tutoren-Korrektur und der Bewertung von Aufgaben
- Übernahme einer studentische Lösung als Musterlösung
- Email an alle Studierenden, einzelne Übungsgruppen und alle Tutoren
- Definieren der Verteilstrategie „Einsendungen → Tutor“. Siehe Abschnitt „Tutoren“.

Studierende

Den Studierende wird auf der personalisierten Einstiegsseite der Lehr-/Lernplattform die zu bearbeitenden Aufgabenhefte mit Abgabetermin angezeigt.
Im Aufgabenbereich können die Studierende die Übungsaufgaben bearbeiten und Zwischenergebnisse abspeichern ohne diese einzusenden. Nach Einsendung der Aufgaben erhalten Sie eine Quittung/Rückmeldung vom System, dass ihre Einsendungen eingegangen sind. Sind vom Betreuer Einsendungen bis zum Einsendeschluss vorgesehen, so können die Studierenden die Einsendungen beliebig oft wiederholen. Das System protokolliert die Einsendungen der Studierenden (Anzahl, Datum). Diese Daten stehen dem Betreuer und dem Dozenten als Statistik zur Verfügung. Müssen die Studierenden Dokumente wie z.B. Winword-Dokumente, Grafiken etc. als Datei einsenden, so haben sie die Möglichkeit die Einsendungen

textuell zu kommentieren. Vor Ablauf der Einsendefrist müssen die Studierenden das Aufgabenheft „virtuell“ abgeben.
Nach Freigabe der Musterlösungen können diese von den Studierenden eingesehen werden, ebenso ihre korrigierten Aufgaben. Jede Korrektur ist mit dem Namen des Tutors versehen, so dass der Studierende seinen Tutor bei Fragen zur Korrektur kontaktieren kann.
Jederzeit kann er sich einen Überblick über seine Einsendungen, Korrekturen und erreichten Punkte machen. Eine persönliche Statistik zeigt ihm an, wie viele Punkte er schon erreicht hat, wie der Punkte-Durchschnitt aller Teilnehmer ist, seine Rangfolge.
Am Ende des Kurses können die Studierenden ihre Aufgaben, Korrekturen, Lösungen. Punkte etc. lokal abspeichern und archivieren kann.

Tutoren

Nach Einsendeschluss werden die Einsendungen automatisch an die Tutoren verteilt. Die Zuordnung „Einsendungen $\rightarrow$ Tutoren“ kann nach den folgenden Strategien durchgeführt:

- Feste Zuordnung. D.h. der Betreuer richtet feste Übungsgruppen ein und legt zu Beginn des Kurses fest, welche Studenten welchem Tutor zugeordnet werden. Jede Übungsgruppe kann über ein Diskussionsforum und ein Dokumentenpool, welche nur für Übungsgruppenmitglieder, Tutor, Betreuer und Dozent zugänglich sind, Informationen austauschen und Aufgaben erarbeiten.
- Aufgabenheftweise nach Stundenzahl der Tutoren
- Aufgabenweise nach Stundenzahl der Tutoren

Die Verteilstrategie ist während des Kurses veränderbar.

Die Tutoren korrigieren die Einsendungen der Studierenden und tragen das Ergebnis in einer Liste ein. Aufgaben, die den Aufgabentypen des Self-Assessments entsprechen, werden vom System automatisch korrigiert und bewertet. Alle anderen Aufgaben werden vom Tutor manuell korrigiert und bewertet.
Der Tutor kann die Korrekturen von Aufgaben entweder online oder offline durchführen. Bei der Offline-Korrektur checkt er seine zu korrigierenden Aufgabenhefte aus und bearbeitet diese lokal. Nach erfolgter Korrektur werden die Korrekturen vom Tutor eingecheckt und die Punkte eingetragen.
Der Tutor kann die textuellen Einsendungen der Studierenden direkt an beliebiger Textstelle annotieren. Bei Dateieinsendungen kann er entweder in einem Feld Kommentare online hinzufügen oder er speichert die Datei lokal auf seinem Rechner ab und kommentiert die Einsendung lokal und führt anschließend ein Datei-Upload durch.
Während der Korrektur werden dem Tutor die maximal zu erreichenden Punktzahl und die Aufgabenstellung angezeigt.
Häufig aufgetretene Fehler kann der Tutor in einen aufgabenspezifischen „Reportpool“ eintragen, auf welchen andere Tutoren, der Betreuer und der Dozent Zugriff haben. An seine Übungsgruppe kann er eine Email versenden.

Fachspezifische Anforderungen:
Informatik:

- Bei Programm-Code kann eine automatische Vorkorrektur durchgeführt werden, die überprüft, ob das Programm fehlerfrei kompiliert wird und korrekte Ergebnisse liefert. Der Programm-Code selbst wird vom Tutor korrigiert und bewertet.

Dozent

Der Dozent konzipiert in Zusammenarbeit mit dem Betreuer die Aufgaben und bespricht nach erfolgter Korrektur die Aufgaben mit den Studierenden. Der Dozent kann wie der Betreuer Reports definieren und erzeugen und sich Statistiken anzeigen lassen.
Im Reportpool kann er sich über häufig aufgetretene Fehler in Aufgabeneinsendungen der Studierenden informieren.
Für jedes Aufgabenheft wird automatisch ein eigener Thread im Diskussionsforum angelegt, über welchen Fragen, Antworten und Informationen zum Aufgabenheft kommuniziert werden können. Um Aufgaben auch synchron mit einer studentischen gruppe zu besprechen, ist eine Erweiterung der synchronen Kommunikation um ein Shared Whiteboard mittelfristig erforderlich.