

Recherche-Bericht

Eine Applikation für den Übungsbetrieb soll das Anmelden der Studenten, diverse Arbeiten wie die Abgabe der Übungsaufgaben übernehmen und eine gewisse Rückinformation über den Leistungsstand bereitstellen. Dazu müssen allerdings bestimmte Daten der Studenten oder der Applikation gespeichert werden. Da die Kommunikation mit dem Benutzer über das Internet von statten gehen soll ist die Entwicklung einer dynamische Web-Applikation von Vorteil und baut somit auf der Grundlage der Java-Servlets auf.

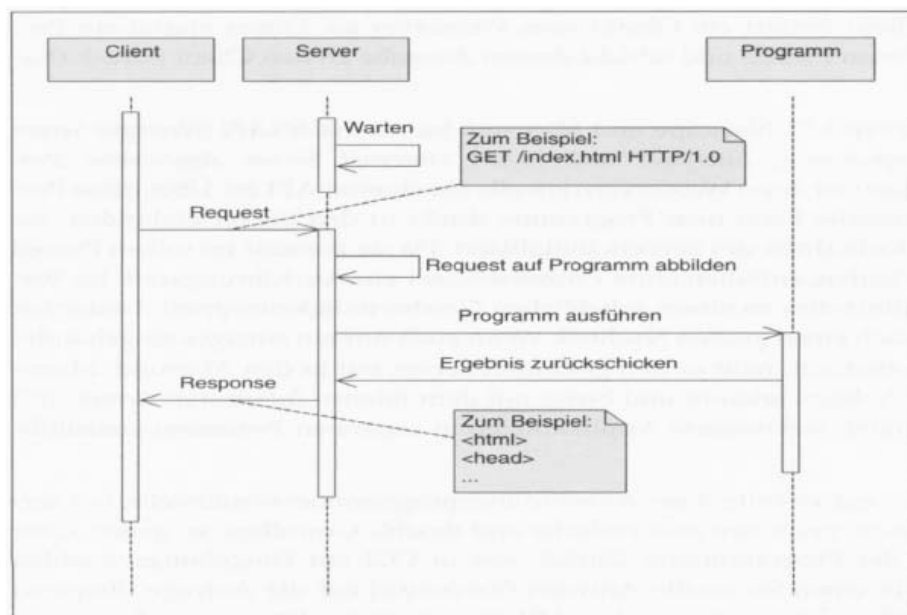
Servlets sind kleine, plattformunabhängige Programmeinheiten, vergleichbar mit den Java-Applets die allerdings nicht auf der Seite des Clients (z.B. im Browser) ablaufen, sondern vielmehr serverseitig ausgeführt werden.

Dabei können diese Servlets die Funktionalität von CGI-Scripts übernehmen und den Ablauf etwa im Bereich von Datenbankabfragen sogar noch beschleunigen. Die Verwendung von Servlets setzt voraus, dass auf dem Server eine Java Virtual Machine (JVM) zur Verfügung steht.

(Bem.: Dies ist etwa dann der Fall, wenn der Java Web Server (JWS, ehemaliger Codename: Jeeves) als WWW-Server eingesetzt wird. Auch der vom World Wide Web Consortium kostenlos vertriebene Beta-Version des JigSaw Web Servers unterstützt Servlets.)

Servlets bieten gegenüber CGI-Scripts verschiedene Vorteile. Wesentlich ist, dass Servlets einen Geschwindigkeitsvorteil bringen können und in geringerem Maß auf System-Ressourcen des Servers zurückgreifen:

Servlets sind speicherresident und müssen daher nur einmal aufgerufen werden. Die plattformunabhängige Einsetzbarkeit der Servlets könnte es noch dazu in Zukunft leichter machen, fertige Lösungen von einem Server zu anderen zu übertragen. Die nebenstehende Grafik stellt die Kommunikation mit einem Servlet dar.



Die Abteilung Informatik der Universität München (LMU) stellt für die Vorlesung „Informatik II“ zur Anmeldung und Einschreibung in den Übungsbetrieb die folgend beschriebene Applikation bereit.



Am oberen Bildrand der Applikation steht dem Studenten ein Menü mit den Auswahlmöglichkeiten Home, Abgabe, Ergebnisse, Benutzerdaten und Anmeldung zur Verfügung. Zur Anmeldung wählt der Student einen Loginnamen und ein Passwort aus. Wenn vorhanden trägt er seine e-mail-Adresse ein. Danach trägt er sich in eine der noch freien Übungsgruppen ein. Die Anzahl der bereit in die Übungsgruppe eingeschriebenen Studenten steht jeweils in der Auswahl. Übungsgruppen, die ihre maximale Gruppenstärke erreicht haben, werden nicht mehr angezeigt und können somit auch nicht mehr ausgewählt werden. Nun trägt der Student noch weitere Informationen wie den Namen, Vornamen, die Matrikelnummer und sein Hauptfach ein, die später zur Erstellung des Übungsscheines benötigt werden. Wenn der Student seinen Übungsschein nicht im Sekretariat

abholen möchte, sondern der Übungsschein per Post zugesandt bekommen möchte, gibt er noch seine Wohnanschrift mit Straße, PLZ und Ort ein.

Dem Studenten steht jederzeit die Möglichkeit seine Benutzerdaten zu ändern zur Verfügung. Nach der Anmeldung kann der Student seine Benutzerdaten auch wieder Ändern um eventuelle Fehleingaben zu korrigieren, sowie seine Aufgaben auf dem Server ablegen und mit einem weiteren Menüpunkt seinen Leistungsstand einsehen.

Aber auch die Abteilung theoretische Informatik – Bereich Berechenbarkeit & Komplexität - der Universität Leipzig bietet eine webbasierte Applikation für den Übungsbetrieb an. Mit Hilfe derer der Student in der Lage ist sich für den Übungsbetrieb anzumelden, seine Aufgaben abzugeben und seinen Leistungsstand abzufragen.

Als erstes muss sich der Student mit seiner Matrikelnummer und ein von ihm gewähltes Passwort anmelden. Danach und bei jedem neuen Einloggen, gelangt der Student auf eine Übersicht auf der ihm seine Anmelde Daten angezeigt werden. Weiterhin wird er auch auf eine eventuell unvollständige Anmeldung hingewiesen. Im unteren Teil der Seite findet der Student ein kleines Menü mit Hilfe dessen er in der Lage ist sich in eine Übungsgruppe einzuschreiben, oder die gewählte zu wechseln. Dazu erscheint auf einer neuen Seite eine Auswahl der noch freien Übungsplätze, Übungsgruppen die ihre maximale Gruppenstärke schon erreicht haben erscheinen nicht mehr. Weiterhin ist der Student in der Lage seine angegebene E-Mailadresse zu ändern, damit das automatische Korrekturprogramm eine Abgabe von Lösungen akzeptiert. Über den vorletzten Menüpunkt kann der Student sein Passwort für die Anmeldung ändern und mit dem letzten Menüpunkt verlässt der Student die Applikation wieder.

Da bei beiden vorgestellten Applikationen die Anzeigedaten auf jeden einzelnen Studenten hin zugeschnitten sind gehe ich davon aus, dass beiden Applikationen die Technik der Servlets Verwendung gefunden hat. Weiter bin ich der Meinung, dass hier eine Einbindung einer SQL-Datenbank nicht von Nöten ist, sondern die abgegebenen Lösungen jedes einzelnen Studenten in einem separaten Verzeichnis untergebracht wurden. Allgemeine Daten wie – Aktuelles, Termine ... - finden hier sicherlich in einem gemeinsamen Ordner ihren Platz. Dies sind allerdings nur Vermutungen über die innere Logik die diese Applikationen beherbergen, welche aus eigenen Überlegungen entstanden sind.

Die typischen Aktionen die eine Applikation für den Übungsbetrieb bereitstellen sollte, ist das selbständige Anmelden eines Studenten. Dabei ist dessen Name, Vorname und seine Matrikelnummer sowie ein von ihm gewähltes Passwort zu registrieren. Dazu werden diese Daten von dem Benutzer in ein erstelltes Formular eingetragen und an das Servlet übergeben. Im Fall der Erstanmeldung werden diese Daten mit in den Datenbestand aufgenommen. Möchte sich der schon angemeldete Student jedoch nur Einloggen um z. B. seine Aufgaben abzugeben, werden die Daten durch das Servlet nur mit dem aktuellen Datenbestand verglichen und somit ein unberechtigter Zutritt verwehrt.

Nachdem sich der Student eingeloggt hat kann er sich in eine Übungsgruppe eintragen oder in eine andere wechseln. Auf Grund dieser Nachfrage wird eine neue Seite durch das Servlet erstellt, auf der nur die Übungsgruppen zur Auswahl stehen die ihre maximale Gruppenstärke noch nicht erreicht haben. Um dies sicherzustellen wird im Datenbestand nachgesehen und die entsprechenden Übungsgruppen zum Beispiel in ein Listefeld eingetragen. Zu beachten ist hier, dass nicht durch die parallele Abarbeitung mehrerer Studenten, das Maximum der Übungsgruppen überschritten wird. Denkbar wäre eine Reservierung eines Übungsplatzes jeder Übungsgruppe beim Aufruf des Auswahlmenüs und eine anschließende Freigabe der nicht gewählten Übungsplätze, sodass kein zweiter Student sich zur gleichen Zeit in dieselbe Übungsgruppe einträgt und deren maximale Gruppenstärke überschreitet.

Möchte der Student nun seine Lösungen abgeben gibt dieser in einem Eingabeformular die entsprechende Übungsserie an und dann wird daraufhin serverseitig in seinem Ordner ein neues Unterverzeichnis angelegt, sofern dies noch nicht durch eine frühere Abgabe erfolgt war. Danach wird per Upload die angegebene Datei in diesem Ordner unter Angabe einer Versionsnummer hinterlegt, sodass im Fall einer Mehrfachabgabe nicht die früheren Versionen überschrieben werden. Zur Korrektur wird jedoch nur die letzte verfügbare Version herangezogen.

Damit sich die Studenten über ihren aktuellen Leistungsstand informieren können, wird eine Liste, aus datenschutzrechtlichen Gründen nur mit Matrikelnummern und der Punktestand aller Studenten

angezeigt. Zusammenfassend werden dabei auch der Durchschnitt und Prozentzahlen ersichtlich. Das Servlet liest dazu eine von den SHK abgelegte Datei mit den Ergebnissen der Studenten aus und erstellt dann daraus eine Übersicht der Leistung der Studenten.

Nachdem nun die studentische Hilfskraft über das Ablaufen der Abgabefrist informiert wurde, soll es ihm möglich sein die Lösungen per Download zu erhalten. Dazu muss auch er sich einloggen und das Servlet erkennt diesen Benutzer als SHK. Durch seine besonderen Rechte kann er nun die Lösungen aller Studenten herunterladen. Dazu gibt die SHK in einem Eingabeformular den Zielort an und dann kopiert das Servlet die Lösungen aller Studenten dorthin. Dabei ist darauf zu achten, dass die Zuordnung der Lösungen zu den Studenten nicht verloren geht.

Nach der Korrektur der Lösungen hinterlegt die SHK eine Datei mit den Ergebnissen auf dem Server, sodass dann das Servlet diese weiter verarbeiten kann.

Damit den SHK und den MA die Arbeit des Übungsbetriebes erleichtert wird soll die Applikation auf Anfrage auch Klarlisten aller Studenten nach Gruppen geordnet ausgeben. Dazu liest das Servlet den aktuellen Datenbestand ein, wertet diesen entsprechend aus und zeigt das gewünschte Ergebnis an.

Wünschenswert ist es wenn die SHK mit Ablauf der Abgabefrist der Übungsserie per E-Mail automatisch benachrichtigt wird, damit sie dann den Download der Lösungen der Studenten vornehmen kann. Auch sollte dem Studenten mitgeteilt werden ob er seine Aufgaben Fristgerecht eingereicht hat oder noch einreichen kann, bei Überschreiten der Abgabefrist könnte dies kenntlich gemacht werden oder die Abgabe gänzlich verweigert werden.

Als nächstes beschreibe ich in tabellarischer Form noch mal die Aktionen die diese Applikation abdecken soll.

Anwendungsfall	Anmeldung des Studenten
Akteur	Student
Beschreibung	Der Student meldet sich mit Namen, Vornamen und Matrikelnummer an und wählt dazu ein Passwort.
Ergebnis	Der Student ist der Applikation bekannt, und kann sich mit seinem Namen und Passwort anmelden.

Anwendungsfall	Wahl der Übungsgruppe
Akteur	Student
Beschreibung	Der Student schreibt sich in eine Übungsgruppe ein, sofern diese noch nicht die maximale Gruppenstärke erreicht hat. Der Upload-Bereich wird freigegeben.
Ergebnis	Der Student wurde einer Übungsgruppe zugewiesen, und kann jetzt seine Lösungen per Upload abgeben. Er erscheint in den Klarlisten geordnet nach Übungsgruppe.

Anwendungsfall	Ändern der Übungsgruppe
Akteur	Student
Beschreibung	Der Student wählt eine neue, noch nicht volle Übungsgruppe. Damit wird er aus der alten entfernt und in der neuen eingetragen. Des Weiteren wird der Platz in der alten Übungsgruppe wieder freigegeben.
Ergebnis	Der Student ist in einer anderen Übungsgruppe eingetragen.

Anwendungsfall	Abgabe der Lösungen
Akteur	Student
Beschreibung	Der Student hinterlegt per Upload seine Lösungen, dabei wird jede Abgabe zur selben Aufgabe nummeriert. Dem Studenten ist die Abgabefrist ersichtlich. Bei erfolgreicher Abgabe wird dem Studenten eine Nachricht über den Erfolg der Abgabe per E-Mail zugesandt.
Ergebnis	Eine oder eine neuere Version einer Lösung des Studenten liegt vor. Und dieser wurde über den Erfolg informiert.

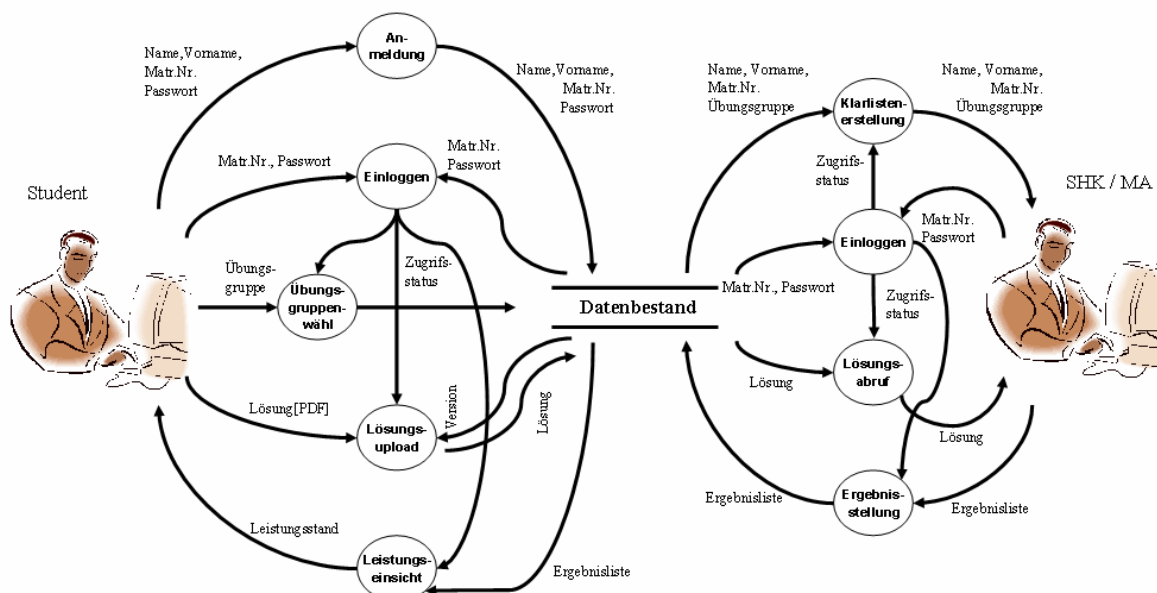
Anwendungsfall	Informieren der SHK über Ablauf der Abgabefrist
Akteur	Keiner
Beschreibung	Die Applikation schickt an die SHK eine E-Mail, sodass sich diese dann die Lösungen der Studenten herunterladen.
Ergebnis	Die Lösungen liegen den Korrektoren bereit.

Anwendungsfall	Download der Lösungen von den SHK
Akteur	Studentische Hilfskraft
Beschreibung	Die SHK lädt sich die Lösungen der Student geordnet nach Übungsgruppe zur Korrektur herunter.
Ergebnis	Die SHK hat die Lösungen der Studenten zur Korrektur vorliegen.

Anwendungsfall	Upload der Ergebnisse
Akteur	Studentische Hilfskraft
Beschreibung	Die SHK hinterlegt eine Liste mit den Ergebnissen auf dem Server ab.
Ergebnis	Die Liste mit den Ergebnissen liegt zur Information der Studenten auf dem Server.

Anwendungsfall	Ansicht der Ergebnisse
Akteur	Student
Beschreibung	Der Student kann sich auf einer öffentlichen Seite über seinen aktuellen Punktestand informieren. Aus datenschutzrechtlichen Gründen werden nur die Matrikelnummern und dazu die erreichten Punkte angezeigt.
Ergebnis	Der Student ist über seinen aktuellen Stand des Übungsbetriebes informiert.

Abschließend noch mal die schon beschriebenen Vorgänge in einem Data-Flow-Diagramm übersichtlich darstellt.



Mitwirkende: Wolfgang Rabe, Matthias Lehmann

Quellen: <http://www.intern.de/97/22/19.shtml>

http://www.galileocomputing.de/openbook/javainsel2/java_170000.htm

<http://www.pst.informatik.uni-muenchen.de/lehre/SS03>

<http://www.informatik.uni-leipzig.de/~joe/edu/ss03/bk/>