

Übung „Netzbasierte Informationssysteme“ WS 2008/2009

Übungsblatt 2

Ausgabe am 04.11.2008

Abgabe bis spätestens 11.11.2008, 14.00 Uhr

Aufgabe 4 - Begrüßung (5 Punkte):

Ziel dieser Aufgabe die Entwicklung einer Begrüßungswebanwendung.

Entwickeln Sie eine statische html-Seite „welcomeDude.html“, die ein Formular enthält. In ein Feld dieses Formulars kann man seinen eigenen Namen eingeben (Bsp: „Thomas Cook“), in das andere kann man den Namen eines Begrüßungsservlets eingeben („GreetingA“ oder „GreetingB“).

Beim Klicken auf einen Button werden die Formulareinträge von einem Management-Servlet „bigChief“ ausgelesen und auf Gültigkeit überprüft. Falls das spezifizierte Begrüßungsservlet nicht existiert (also weder „GreetingA“ noch „GreetingB“ entspricht, leitet „bigChief“ die response auf eine statische Ergebnisseite „error.html“ weiter. Diese soll auf die fehlerhafte Eingabe hinweisen.

Bei korrekter Eingabe reicht das Management-Servlet den request an das spezifizierte Begrüßungsservlet weiter. Implementieren Sie zwei Begrüßungsservlets – „GreetingA“ und „GreetingB“. Diese nehmen den request vom „bigChief“ entgegen und generieren eine html-Ausgabe der Art: „nameDesBegrüßungsservlets says hello to nameOfUser!“. Bsp: „GreetingA says hello to Thomas Cook!“.

Entwickeln Sie Ihre Sourcen in Eclipse in einem neuen folder aufgabe4/src. Deployen Sie Ihre Webanwendung mittels eines ANT build.xml scripts. Testen Sie Ihre Webanwendung!

Aufgabe 5 – GURU (2 Punkte):

Auf der Vorlesungshomepage finden Sie zwei einfach JSP Beispiele: GuruScriptlet.jsp und GuruTagged.jsp die eine Guru.java Hilfsklasse verwenden.

Deployen Sie diese beiden Webanwendungen unter /webapps/aufgabe5 und testen Sie die Funktionalität. Gehen Sie den Sourcecode durch und vergleichen Sie die beiden Arten, ein JSP zu erstellen.

Wechseln Sie in das /work – Verzeichnis Ihres lokalen Tomcats und suchen Sie nach den aus den JSPs erzeugten Servlets. Kopieren Sie den Code der Servlets in Textfiles in Ihre Eclipse-Umgebung: aufgabe5/src. Markieren Sie den Abschnitt in den Servlets wo jeweils die Ergebnis-html Seite erstellt wird. Fügen Sie dazu Sternchen (****) vor und nach den Absätzen ein.

Aufgabe 6 – Airlineverwaltung (5 Punkte):

In dieser Aufgabe soll eine Airlineverwaltungs-Webanwendung entwickelt werden. Bilden Sie mittels Javaklassen folgendes Szenario ab (package „airlinemgt“):

Ein Flugzeug hat eine Benennung, einen Flugzeugtyp und eine bestimmte Sitzanzahl. Ein Flug hat eine Flugnummer, ein Flugdatum, eine Abflug- und Ankunftszeit sowie einen Start- und Zielort (bilden Sie der Einfachheit halber alle Attribute der Klassen als Strings ab).

Da die Airline noch rückständig ist und keine Datenbank besitzt entwerfen Sie eine Flugverwaltungsklasse (AirlineAdmin). Diese soll die Flug- und Flugzeugobjekte speichern. Welche Datenstruktur Sie dabei verwenden, bleibt Ihnen überlassen.

Ihr Chef würde gerne das neue Flugverwaltungssystem mittels eines Browsers administrieren. Helfen Sie ihm dabei! Entwickeln Sie zuerst eine statische Einstiegsseite (welcome.html) von der aus die anderen Seiten via Link zu erreichen sind. Je ein Link soll zu einem Servlet führen, in dem mittels eines Formulars neue Flug- bzw. neue Flugzeuginstanzen erzeugt werden (CreateFlight, CreatePlane).

Um Sessionpersistenz der eingetragenen Flüge und Flugzeuge zu erreichen, legen Sie (falls noch nicht vorhanden) eine Instanz Ihrer entwickelten Flugverwaltungsklasse an und binden Sie diese als Attribut in den Sessionkontext ein. In dieser Instanz speichern die beiden Servlets neu erzeugte Flugzeug bzw. Fluginstanzen.

Natürlich will Ihr Chef auch einen Überblick über den momentanen Inhalt des Flugverwaltungssystems erhalten. Entwerfen Sie dazu ein JSP, das die gespeicherten Flugzeuge und Flüge sowie deren Daten in Tabellenform übersichtlich darstellt (Cockpit.jsp). Dieses soll auch via Link von der welcome-Seite erreichbar sein.

Deployen Sie die Webanwendung und testen Sie die Funktionalität! Fangen Sie unzulässige Zustände ab, beispielsweise soll das JSP einfach leere Tabellen anzeigen falls keine Flugverwaltungsinstanz im Kontext liegt oder deren Inhalt leer ist. Zusätzliche Exceptions sollten Sie abfangen und bei deren Auftreten auf eine statische error.html Seite leiten.

(Optional für interessierte Code-Gurus, 2 Bonuspunkte:

Ermöglichen Sie zusätzlich online nach einer Methode Ihrer Wahl (Servlet, JSP) die Zuordnung von jeweils genau einem der gespeicherten Flugzeuge zu den angelegten Flügen. Diese Zuordnung soll in der Instanz der Flugverwaltungsklasse gespeichert werden. Integritäten wie „ein Flugzeug darf nicht gleichzeitig für zwei Flüge eingeteilt werden“ oder „mögliche Reihenfolge der Flüge in Bezug auf Startflughafen und Endflughafen“ oder ähnliches werden vom bedienenden Airlinepersonal beachtet, müssen also nicht von Ihnen behandelt werden. Zeigen Sie die neuen Zuordnungen von Flugzeugen zu Flügen auch innerhalb des bereits entwickelten JSPs in Tabellenform an.)

Aufgabe 7: HTML Webseitenerstellung und Veröffentlichung (3 Punkte)

a) Website erstellen (1 Punkt)

Es soll eine Website entstehen, die 10 unterschiedliche HTML Seiten beinhaltet. Wie diese Seiten miteinander verlinkt sind, ist Ihnen überlassen. Jedoch soll eine Seite als "Hauptseite" fungieren (diese könnte beispielsweise 'index.html' heißen) und jede Seiten soll mindestens einen eingehenden Link von einer anderen Seiten haben.

Die HTML Seiten können beliebigen textuellen Inhalt haben, allerdings muss er ohne Copyright sein. Wikipedia bietet sich insbesondere als Quelle für diesen Inhalt an.

Die HTML Seiten müssen gültige HTML Markups enthalten. Zumindest sollte HTML, HEAD und BODY Tags sowie P, BR, H1, H2, H3, B, I, U usw. für die textuelle Formatierung verwendet werden.

Auf wenigstens einer Seite sollen mindestens drei Adressen in XML Format angegeben werden. Sie können Ihr eigenes XML Vokabular dafür entwickeln oder ein schon existierendes Vokabular verwenden. Außerdem muss ein Namensraum benutzt werden (xmlns Attribut auf das Wurzelement).

Mindestens fünf Seiten sollen wenigstens ein Bild (IMG tag) enthalten. Die Bilder müssen ohne Copyright sein. Wikimedia Commons bietet sich insbesondere als Quelle für Copyright-freie Bilder.

Die Website sollte alle Good Practices berücksichtigen. (siehe Architecture of the World Wide Web, Volume One: List of Principles, Constraints and Good Practice Notes)

b) Veröffentlichung auf lokalem Webserver (1 Punkt)

Sie können die Interaktion mit Ihrer Website über das Internet simulieren, indem Sie lokal einen Web Server installieren. Dafür eignet sich Apache Tomcat: Fügen Sie in einer Tomcat Installation die Daten Ihrer Website inklusive einer Indexseite (index.html) ein, die entweder die Hauptseite ihrer Website ist oder eine Umleitung (HTML Redirect) darauf macht. Konfigurieren Sie die Webanwendung mit dem Kontextpfad (context path) /aufgabe7. Aufgabe 7b ist erledigt, sobald Sie über <http://localhost:8080/aufgabe7> die Indexseite Ihrer Website aufrufen können.

c) Website online stellen (1 Punkt)

Die Website soll auch online auf den Fachbereichserver mi.fu-berlin.de gestellt werden. Dafür verwenden Sie Ihren privaten Homepagebereich. Bitte schicken Sie in der Abgabe die URL auf die von Ihnen veröffentlichte Seite zu.