

PI.SWA.SA.VO Software Architekturen und Web Technologien

3. Termin WS 2007
Schriftliche Einzelprüfung
Di 23.04.2008

Prüfer: Derntl/Motschnig

Anweisungen:

- **Arbeitszeit: 90 Minuten**, es sind **max. 90 Punkte** zu erreichen (die Anzahl der Punkte, die pro Aufgabe zu erreichen ist, finden Sie jeweils in eckiger Klammer). Kalkulieren Sie also für jeden erreichbaren Punkt ca. eine Minute Arbeitszeit.
- Halten Sie einen **Lichtbildausweis** (Studentenausweis, Führerschein, Pass, etc.) bereit.
- Lösen Sie die Aufgaben direkt auf den Angabeblättern und schreiben Sie **leserlich**.
- Falls Sie zusätzliche Blätter benötigen, fragen Sie die Prüfungsaufsicht.
- Es sind **keine Unterlagen**, Handies, PDA, Laptops, Taschenrechner, Headsets, oder sonstige Hilfsmittel erlaubt!

----- (zum Abtrennen und Mitnehmen) -----

Hinweise:

- Ergebnisse werden bis spätestens Di 13.5. im PISWI verfügbar sein.
- **Einsicht ist am Mi 14.5. von 10–11 Uhr** im Dienstzimmer Derntl, Rathausstrasse 19, 1. Stock, Tür 9
- Dies ist der 3. von insgesamt 4 Terminen aus dem Stoffsemester WS 2007. Ein weiterer Termin folgt im Juni 2008.

Aufgabe 1: XML Familie**[40]**

Gegeben ist folgendes XML Dokument mit Informationen über Spielfilme:

```
<?xml version="1.0"?>
<movies>
  <movie id="tt0065214" year="1969">
    <title>The Wild Bunch</title>
    <director>Sam Peckinpah</director>
    <genre>Action</genre>
    <genre>Drama</genre>
    <genre>Western</genre>
  </movie>
  <movie id="tt0083658" year="1982">
    <title>Blade Runner</title>
    <director>Ridley Scott</director>
    <genre>Drama</genre>
    <genre>Sci-Fi</genre>
    <genre>Thriller</genre>
  </movie>
  <movie id="tt0069762" year="1973">
    <title>Badlands</title>
    <director>Terrence Malick</director>
    <genre>Crime</genre>
    <genre>Drama</genre>
    <genre>Romance</genre>
    <genre>Thriller</genre>
  </movie>
  ... weitere Filme ...
</movies>
```

- 1a. Formulieren Sie ein oder mehrere **XSL Templates**, mit dem/denen man die obige XML Filmdatenbank in eine HTML Liste mit folgendem Format transformieren kann:

```
<ul>
  <li><b>The Wild Bunch</b>: Sam Peckinpah, 1969</li>
  <li><b>Blade Runner</b>: Ridley Scott, 1982</li>
  <li><b>Badlands</b>: Terrence Malick, 1973</li>
  ... weitere Filme ...
</ul>
```

Anmerkung: Zu jedem Film sollen also der Filmtitel (**title**) in Fettschrift, der Regisseur (**director**) und das Jahr (**year**) in dem dargestellten Format ausgegeben werden. [12]

1b. Formulieren Sie jeweils einen **XPath** für folgende Abfragen auf die XML Filmdatenbank:

- i. Alle Filme vor dem Jahr 2000, die nur einem einzigen Genre zugeordnet sind, und deren Regisseur entweder "Sergio Leone" oder "«VORNAME» «NACHNAME»" war. [3]

- ii. Alle Filme, deren Regisseur mehr als einen Film gedreht hat. [3]

1c. Gegeben ist folgende DTD:

```

<!ELEMENT r (x|y|z)>
<!ELEMENT x ((a|b)+|c*)>
<!ELEMENT y (a?,b)+>
<!ELEMENT z (#PCDATA)>
<!ELEMENT a EMPTY> <!ELEMENT b EMPTY> <!ELEMENT c EMPTY>
  
```

In der folgenden Tabelle steht jede Zeile für ein eigenes XML Dokument. Kreuzen Sie in jeder Zeile an, ob das XML *gültig* oder *ungültig* für die gegebene DTD ist! [12]

	gültig	ungültig
<r><y></r>		
<r><z></r>		
<r><x></x></r>		
<r><z>Hello, my <dear></z></r>		
<r><x><a></x></r>		
<r><x><c><c><c></x></r>		
<r><x><y></x></r>		
<r><x></r>		
<r><y><a></y></r>		
<r><z>Hallo</z></r>		
<r><z><a></z></r>		
<r><x><a><c></x></r>		
<r><y><a></y></r>		
<r><y>zwei noch</y></r>		
<r><y></y></r>		
<r><x></x></r>		

- 1d.** Erläutern Sie anhand einer kleinen XML Beispieldatei den Zweck und das Konzept von **XML Namespaces** (Anmerkung: nicht nur das XML niederschreiben, sondern auch in eigenen Worten erläutern!) [10]

Aufgabe 2: Software-Architekturen

[35]

2a. Erklären Sie die folgenden Software **Architekturprinzipien** jeweils kurz. Gehen Sie dabei auch auf die Zusammenhänge zwischen den Prinzipien ein:

i. Abstraktion: [3]

ii. Separation of concerns: [3]

iii. Information Hiding: [3]

iv. Modularität: [3]

2b. Architektursichten:

i. Was versteht man unter Architektursichten, und wie werden sie beschrieben? [3]

ii. Wie hängen Architektursichten und Architekturmodell zusammen? [1]

iii. Erläutern Sie drei der fünf Sichten des *Abstrakten Architektur-Modells*: Geschäftssicht, Logische Sicht, Datensicht, Realisierungssicht, Verteilungssicht. [9]

2c. n-Tier Architekturen:

- i. Was versteht man unter einer n-Tier Architektur? Aus welchen Schichten besteht diese üblicherweise? [4]

- ii. Skizzieren Sie Schichtenanordnungen für einen *fat client* und einen *thin client*. [3]

- iii. Vergleichen Sie n-Tier Architekturen mit dem Layers Pattern. [3]

Aufgabe 3: Webanwendungen

[15]

- 3a. Erläutern Sie das **AJAX** Konzept. Denken Sie sich ein Einsatzbeispiel aus, wo die Verwendung von AJAX Sinn machen würde. [6]

- 3b.** Was sind HTTP **Cookies** und wozu werden sie benötigt? Wie sehen der Ablauf und die Rollen bzw. Aufgaben von Client, Webapplikation und HTTP-Protokoll bei der Verwendung von Cookies aus? [9]