

Otto-Friedrich-Universität Bamberg

Modulprüfung

- Teilprüfung:** Web-Technologien (MI-WebT-B)
- Fach:** Medieninformatik
- Aufgabensteller:** Prof. Dr. Andreas Henrich
- Prüfungstermin:** Sommersemester 2017
- Bearbeitungszeit:** 90 Minuten (Zusätzlich wird eine Lesezeit von 15 Minuten gewährt.)
- Hilfsmittel:** nicht programmierbarer Taschenrechner
- Hinweise:** Setzen Sie in der Kopfzeile der einzelnen Blätter jeweils Name und Matrikelnummer ein. Bearbeiten Sie die Aufgaben unter Berücksichtigung der in den Klammern vermerkten maximal erreichbaren Punktzahl.
- Wenn Annahmen gemacht werden müssen, sind diese zu begründen. Schreiben Sie gut leserlich, unleserliche Teile werden nicht gewertet.
- Beachten Sie, dass bei der Punkteverteilung **ein Punkt einer erwarteten Bearbeitungsdauer von ca. einer Minute entspricht**. Sie sollten also z. B. die Antwort auf eine Aufgabe mit 5 Punkten so bemessen, dass zur Beantwortung 5 Minuten ausreichen.

Wichtiger Hinweis:

Es sind 7 Aufgaben á 15 Punkte gegeben, von denen 6 zu bearbeiten sind. Die maximal erreichbare Punktzahl beträgt somit 90 Punkte!

Bearbeiten Sie alle 7 Aufgaben, gehen nur die 6 Aufgaben, bei denen Sie die höchsten Punktzahlen erreicht haben, in die Gesamtpunktzahl ein! D. h., die Punkte für die Aufgabe, bei der Sie die geringste Punktzahl erreicht haben, werden nicht berücksichtigt!

Viel Erfolg!

Name des Studenten / der Studentin:

----- *ab hier ist diese Seite vom Prüfer auszufüllen* -----

Punkte aus der Klausur:

Aufgabe 1	Aufgabe 2	Aufgabe 3	Aufgabe 4	Aufgabe 5	Aufgabe 6	Aufgabe 7	Summe

zuzüglich _____ Punkte aus den Studienleistungen (sofern Klausur bestanden): _____

Note: _____

Aufgabe 1 (Grundlagen und Vermischtes, 15 Punkte):

- a) Erläutern Sie den Aufbau einer URL. Gehen Sie dabei auf die syntaktische Zusammensetzung und die Bedeutung der einzelnen Bestandteile ein. Erklären Sie in diesem Zusammenhang ebenfalls, was sich hinter dem Begriff DNS verbirgt und wie er mit der URL in Verbindung steht. (6 Punkte)
- b) Erläutern Sie, wie die Typisierung auszutauschender Daten mit dem MIME-System arbeitet. (4 Punkte)
- c) Eine wichtige Eigenschaft von HTTP ist es, *zustandslos* zu sein. Was bedeutet dies und warum erscheint es sinnvoll? Insgesamt kommen Web-Anwendungen und Protokolle, die beim Datenaustausch genutzt werden, allerdings nicht ohne Zustände aus. Geben Sie die verschiedenen Stellen an, an denen Zustände verwaltet werden, und begründen Sie kurz, warum dies erforderlich ist. (5 Punkte)

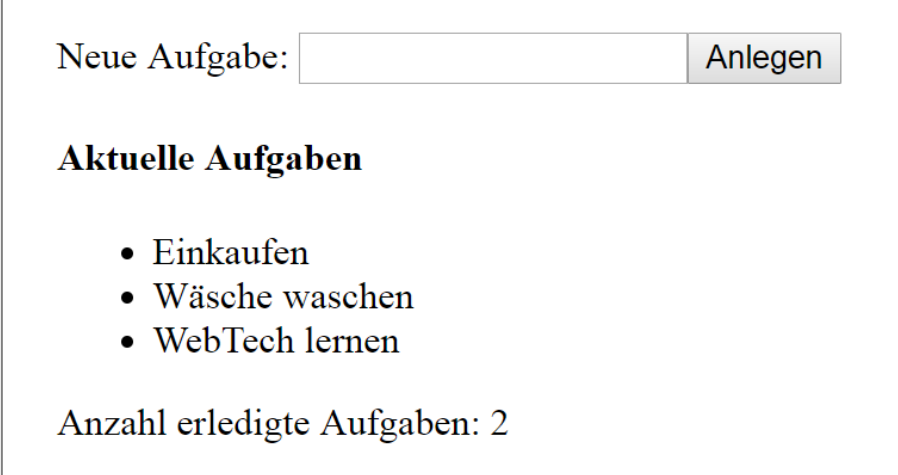
Aufgabe 2 (HTML5 und CSS3, 15 Punkte):

- a) Ein Freund hat Sie gebeten, ihm beim Einbetten des Videos *jQuery_intro.mp4* auf einer HTML5-Seite zu helfen. Leider konnte Ihnen Ihr Freund nicht sagen, in welchem Ordner auf dem Webserver die HTML-Seite letztendlich liegen soll. Sie wissen aber, dass sich das besagte Video in einem Verzeichnis *vid* befindet. Dieses Verzeichnis befindet sich direkt im Wurzelverzeichnis des Servers.
Geben Sie ein HTML5 Code-Snippet an, das das gewünschte Video einbindet. Die Wiedergabe soll gleich beim Seitenaufruf starten, außerdem soll der Nutzer auch die Möglichkeit haben, das Video zu pausieren. Ihr Freund möchte, dass möglichst alle Besucher seiner Seite das Video sehen können. Welchen Hinweis sollten Sie Ihrem Freund zu dieser Anforderung noch geben? (4 Punkte)
- b) Farben sind essentielle Bestandteile fast aller Layouts. Geben Sie zwei grundsätzliche Möglichkeiten an, mit denen Farben im CSS-Code festgelegt werden können. Nennen Sie auch zwei verschiedene CSS-Eigenschaften, bei denen Farben festgelegt werden müssen. (2 Punkte)
- c) Viele professionelle Webentwickler nutzen anstatt von CSS andere Stylesheet-Sprachen, wie z.B. *Less* oder *Sass*. Erläutern Sie zwei Features, die von solchen Sprachen unterstützt werden und die bei CSS fehlen. (3 Punkte)
- d) Geben Sie für jeden der drei folgenden CSS-Selektoren ein **minimales** HTML-Beispiel an, sodass jeweils mindestens ein Element selektiert wird und verdeutlichen Sie dieses. Beginnen Sie jeweils mit dem `<body>`-Element. (6 Punkte)
 - `h1, h2 span {...}`
 - `table tr:nth-child(3) td#warning {...}`
 - `article[title="Simpsons"] div[title] {...}`

Aufgabe 3 (JavaScript und jQuery, 15 Punkte):

Die Abbildung zeigt den Screenshot einer einfachen ToDo-Liste, die mit HTML und JavaScript umgesetzt wurde: Um einen neuen Eintrag anzulegen, gibt man die Bezeichnung der Aufgabe ins Textfeld ein und klickt auf den **Anlegen**-Button. Daraufhin wird das Textfeld geleert und die neue Aufgabe der Liste hinzugefügt.

Bei einem Klick auf ein Listenelement wird dieses aus der Liste entfernt und der Zähler unten auf der Seite für die Anzahl der erledigten Aufgaben um eins erhöht.



The screenshot shows a web form for a To-Do list. At the top, there is a label 'Neue Aufgabe:' followed by a text input field and a button labeled 'Anlegen'. Below this, the heading 'Aktuelle Aufgaben' is displayed. Under the heading is a bulleted list with three items: 'Einkaufen', 'Wäsche waschen', and 'WebTech lernen'. At the bottom of the form, it says 'Anzahl erledigte Aufgaben: 2'.

Ihre Aufgabe ist es, die Funktionalität der beschriebenen Webseite möglichst gut nachzubilden. Geben Sie dazu zuerst geeigneten **HTML**-Code an (der Inhalt des body-Elements genügt). Geben Sie danach passenden **JavaScript**-Code an. Sie können dabei davon ausgehen, dass im Code auch die **jQuery**-Bibliothek eingebunden wird.

Hinweis: Um einem selektierten Element ein Kindelement hinzuzufügen, können Sie die *append*-Funktion nutzen.

Aufgabe 4 (Server-Side Scripting, 15 Punkte):

Sie sollen hier einen Einkaufszettel als **PHP-Anwendung** schreiben, die zur Verwaltung des Zettels das **Session-Konzept** nutzt.

Beim ersten Aufruf soll die Anwendung einen leeren Zettel liefern:

Einkaufszettel

Neue Position hier eingeben:

Gewünschte Menge:

Einheit angeben:

Bisher auf dem Einkaufszettel:

Der Einkaufszettel ist leer!

Nach 3 Eintragungen könnte der Zettel z. B. wie folgt aussehen:

Einkaufszettel

Neue Position hier eingeben:

Gewünschte Menge:

Einheit angeben:

Bisher auf dem Einkaufszettel:

- Butter (250 Gramm)
- Schinken (4 Scheiben)
- Milch (2 Liter)

Die umseitig angegebene PHP-Datei `einkauf.php` gibt den Rahmen für Ihre Anwendung vor. Schreiben Sie PHP-Code für die beiden ausgelassenen Blöcke.

Eine genaue Fehlerbehandlung und Sicherheitsaspekte können hier unbeachtet bleiben.

Beachten Sie bitte, dass zur Bearbeitung der Aufgabe nur 15 Minuten vorgesehen sind. Auf Teillösungen werden anteilige Punkte vergeben.

[Template für `einkauf.php` auf der Folgeseite]

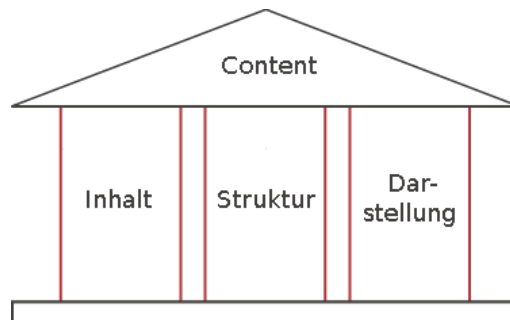
```
1 <?php
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18 ?>
19
20 <!DOCTYPE html>
21 <html>
22 <head>
23 <meta charset="UTF-8"/>
24 <title>Einkaufsliste</title>
25 </head>
26 <body>
27
28 <h1>Einkaufszettel</h1>
29
30 <form action="einkauf.php" method="post">
31 <table>
32 <tr><td><label for="posten">Neue Position hier eingeben:</label></td>
33 <td><input type="text" name="posten"></input></td></tr>
34 <tr><td><label for="menge">Gewünschte Menge:</label></td>
35 <td><input type="text" name="menge"></input> </td></tr>
36 <tr><td><label for="einheit">Einheit angeben:</label> </td>
37 <td><input type="text" name="einheit"></input></td></tr>
38 <tr><td></td><td><input type="submit" value="speichern"></input></td></tr>
39 </table>
40 </form>
41
42 <h2>Bisher auf dem Einkaufszettel:</h2>
43 <?php
44
45
46
47
48
49
50
51
52
53 ?>
54 </body>
55 </html>
```

Aufgabe 5 (Webentwicklung, Frameworks, CMS, SEO 15 Punkte)

- a) Caching ist an verschiedenen Stellen in einer Web-Anwendung möglich. Skizzieren Sie in der folgenden Abbildung (durch Markierung mit ①, ② und ③) 3 verschiedene Stellen, an denen Caching möglich ist und diskutieren Sie unter Bezug auf diese Stellen auf dem Lösungsbogen kurz die Vor- und Nachteile sowie welche Nutzer jeweils vom Caching an der Stelle profitieren könnten. (6 Punkte)



- b) Die folgende Abbildung aus WinfWiki (<http://winfwiki.wi-fom.de/index.php/Hauptseite>) zeigt die „drei Säulen des Content“ auf. Erläutern Sie, wie diese Säulen in einem WCMS umgesetzt und unterstützt werden. (5 Punkte)



- c) Ihr Chef will eine kurze Handreichung zu SEO für die Web-Abteilung im Hause verfassen lassen. Darauf sollen in einer Auflistung vier einfach anwendbare Merksätze das Wichtigste zum SEO festhalten. [4 Sätze müssen es wohl sein, weil er am 4.4. Geburtstag hat.] Schreiben Sie eine solche Handreichung. (4 Punkte)

Aufgabe 6 (Play Framework, 15 Punkte):

- a) Betrachten Sie die URL „*http://localhost:9000/planttype/delete/5*“ und geben Sie dann eine mögliche, syntaktisch korrekte Route für diesen Aufruf für die *routes* Datei an. (2 Punkte)
- b) Beschreiben Sie detailliert den Ablauf der Anfrageverarbeitung für den Aufruf der URL aus Teilaufgabe 6 a). Ihre Schilderung sollte dabei mit dem Aufruf einer Methode in einer Controller-Klasse gemäß den Einträgen aus der *routes* Datei beginnen. Beziehen Sie bei Ihrer Beschreibung auch Datenbankzugriffe mit ein und gehen Sie davon aus, dass der Aufruf das Löschen einer angelegten Pflanzenart bewirkt. (5 Punkte)
- c) Sie wurden damit beauftragt, ein sehr kompaktes Informationssystem für Ihre Universität zu entwickeln und haben sich für eine Umsetzung mit dem Play Framework entschieden. Aktuell muss folgender Sachverhalt in der Datenbank abgebildet werden:

Studierende verfügen über Vor- und Nachname (wobei der Vorname keine Pflichtangabe ist) und können an mehreren Modulen teilnehmen. Für ein Modul muss verpflichtend die Modulbezeichnung und das Datum der ersten Veranstaltung angegeben werden. An einem Modul können natürlich mehrere Studierende teilnehmen. Natürlich müssen sowohl Studenten als auch Module eindeutig identifizierbar sein.

Geben Sie vollständigen Code für die beiden Model-Klassen *Student* und *Modul* an, *package*- und *import*-Anweisungen können weggelassen werden. (8 Punkte)

Aufgabe 7 (Sicherheit, 15 Punkte):

- a) Erläutern Sie jeweils kurz die fünf in der Vorlesung behandelten Schutzziele. (5 Punkte)
- b) Was versteht man unter „Persistentem XSS“? Wie ist der grundsätzliche Ablauf? Welche Maßnahmen können und sollten bei der Entwicklung einer Web-Anwendung getroffen werden, um dieser Bedrohung zu begegnen? (4 Punkte)
- c) Der folgende Ausschnitt einer Webseite möchte den Benutzer nach einen Aufruf mit dessen Namen begrüßen. Dazu wird der Name beim Aufruf (z.B. ausgelöst durch ein Formular) als Parameter übergeben. Ein Beispielaufruf könnte wie folgt aussehen:

```
http://www.beispiel.de/index.html?name=Andreas
```

Der übergebene Parameter wird nun in der Seite `index.html` aus der URL ausgelesen und in die Webseite integriert:

```
Hallo
<script>
  var pos = document.URL.indexOf("name")+5;
  document.write(document.URL.substring(pos,document.URL.length));
</script>
,<br/>
Herzlich Willkommen auf dieser Seite ...
```

Bei obigem Aufruf würde dies zu folgender Ausgabe führen:

```
Hallo Andreas,
Herzlich Willkommen auf dieser Seite ...
```

Gegen welche Art von Angriff ist die Datei `index.html` anfällig? Wie könnte ein Angriff aussehen? Wie könnte man einen solchen Angriff verhindern? (6 Punkte)

[Sie können hier ignorieren, dass verschiedene Browser sehr direkte Nutzungen der Sicherheitslücke durch entsprechende Kodierung der URL ggf. bereits unterbinden würden. (Quelle: <https://www.ceilers-news.de/...>)]