

Backend Basics

REST-Full Service mit PhP

Agenda

- Diskussion zur Übung 3
- REST-Full Service
- Besprechen und Planen der Übung 4

REST-Full Services

REST: Representational State Transfer

Warum?

- Architekturstil der dem WWW entspricht (HTTP Requests per URL)
- einheitliche Schnittstelle (im Vergleich SOAP, WSDL - APP abhängig)
- Austausch von Information zwischen zwei Applikationen

REST: Representational State Transfer

Wie?

- Nutzen der vorhandenen HTTP Infrastruktur des WWW
- Angabe des Ortes der Funktionalität per URI (kein Methodenaufruf)
- Zustandslosigkeit als Prinzip (aber: SESSION, COOKIES)

REST: Representational State Transfer

Was?

- HTTP(S) Request zur Kommunikation
- Datenübertragung durch HTTP Methoden: GET, POST, PUT, DELETE, PATCH, HEAD, OPTIONS, CONNECT, TRACE
- JSON als standardisierte Datenstruktur
- SESSIONs und COOKIES als Wiedererkennung des Clients

POST vs. GET

- GET (in php: `$_GET`)
 - `https://example.com?GETPARAM1=value1&GETPARAM2=value2`
 - URLs sind de facto auf ca. 2.000 Zeichen je nach Browser beschränkt
 - laut RFC größer
- POST (in php: `$_POST`)
 - Parameter werden im HTTP Envelope übertragen
 - maximale Größe kann vom Server vorgegeben werden
 - Übertragung von Dokumenten, Bildern, etc.

COOKIES

- um den Browser / Client zu identifizieren
- Browser mitteilen, er soll ein Cookie setzen
- Datei wird angelegt

```
<?php
//http://php.net/manual/de/function.setcookie.php
setcookie("TestCookie", $value, time()+3600, "/~rasmus/", "example.com", 1);

// ein bestimmtes Cookie ausgeben
echo $_COOKIE["TestCookie"];

// Ein anderer Weg zu Debuggen/Testen ist, alle Cookies anzuzeigen
print_r($_COOKIE);
```


SESSIONS

- den Status des Clients für spätere Requests am Server merken
- Apache Server mitteilen einen Eintrag in die Session zu schreiben
- Datei am Server enthält Session
- ungenutzte Sessions laufen nach gewisser Zeit ab => Daten unverfügbar
- Reihenfolge beim Arbeiten mit Sessions der Aufrufe kann tricky sein

Sessions

- klassischer Anwendungsfall: Login

```
<?php
//http://php.net/manual/de/function.session-start.php
session_start();

// einen Session Parameter setzen. z.B. die echte ID auf der DB (geheim)
$_SESSION["userId"] = $databaseUserId;

// gesamte Session betrachten
print_r($_SESSION);

// Session Variable löschen
unset($_SESSION["userId"]);

// gesamte Session auflösen
session_destroy();
```

Fragen bis hier?

Übung 4

Planung und Implementierung: Erweitern Sie die Übung mit den Produktlisten (gerne Ihren eigenen Code, oder die Musterlösung) mit der Funktionalität eines Warenkorbes. Der Client soll über zwei Schnittstellen jeweils einen Artikel hinzufügen können, entfernen (mehrfach möglich: n mal hinzu oder weg => die Anzahl wird je Eintrag mehr oder weniger) und mit einer weiteren Schnittstelle die Liste der Waren im Warenkorb anzeigen können. Verwenden Sie weiterhin die MVC Architektur und überlegen Sie welche Technologie zum behalten des Warenkorbinhalts eingesetzt werden kann.

Schnittstellen

- `http://localhost/Uebung4/index.php?action=addArticle&articleId=10`
- `http://localhost/Uebung4/index.php?action=removeArticle&articleId=10`
- `http://localhost/Uebung4/index.php?action=listCart`

Output:

- für add und remove: `{"state": "OK / ERROR"}`
- listCart:

```
{ cart:
  [ {"articleName": <string>, "amount": <int>}, ... ]
}
```

Abgabe der Übung 4

- bis 31.05.2020
- im Moodle unter Uebung 4
- NACHNAME_Uebung4.zip (kein *.rar)
 - mit allen Files des Projektes
 - bitte keine versteckten Files die sonst in Ihrem Filesystem vorkommen *.ts, *.ps, usw.
 - Ihr Entwurfskonzept, wie üblich als Scan/Foto/etc.
- Fragen bitte im Diskussionsforum (Keine Scheu! Wir sind hier um zu lernen!)
- versuchen Sie es wirklich alleine
- vergleichen Sie mit der Anwendung aus den vorigen Übungen
- versuchen Sie clean zu Implementieren

Planen der Anwendung

- überlegen Sie welche Objekte wahrscheinlich eine Rolle spielen werden
- wie könnten die Namen der Objekte lauten
- welche Eigenschaften und Methoden werden die Objekte brauchen
- wo wird es notwendig sein Schleifen oder Fallunterscheidungen einzuplanen
- welche Parameter müssen weiter gereicht werden?
- **zeichnen Sie einen leserlichen Plan auf**