

Web-basierte Anwendungen 2: Verteilte Systeme

Gruppe 1 / Team 9

Petya Ivanova
Dominik Bronislaw Schliewowski
Jan Oliver Pietrek

Grobkonzept

Rent @ Bike

Inhaltsverzeichnis

Einleitung	3
Ressource.....	3
Topics	5
Datenstrukturen.....	5
Kommunikationsmodel.....	7

Einleitung

Mit dem Grobkonzept beschreiben wir die Grundvorstellung unseres Projektes und es dient zur Sammlung unserer Leitgedanken.

Das Konzept stellt dar wie ein Mensch auf eine einfache Art und Weise, Fahrräder leihen und verleihen kann. Anhand dessen ist ein grobes Ziel zur Lösung des Problems entstanden. Um dieses zu erreichen müssen Anforderungen ermittelt werden. Anhand der Anforderungen wurde recherchiert ob es bereits Systeme gibt die ähnlich sind.

Ressourcen

GET:

Die GET-Methode ist in der Lage, Informationen jeglicher Art mithilfe der Ergebnis-URI zu identifizieren. GET ist die gebräuchlichste Methode. Mit ihr wird eine Ressource (z. B. eine Datei) unter Angabe eines **URI** vom Server angefordert. Als Argumente in dem URI können also auch Inhalte zum Server übertragen werden. ¹

Systembezogen:

Der Benutzer bezieht seine Profildaten über eine GET - Methode. Das Benutzerprofil als solches, liegt auf einem externen Medium, z.B. Datenserver oder Webserver.

¹ <http://www.selfphp.info/praxisbuch/praxisbuchseite.php?site=183&group=32>

PUT:

Die PUT – Methode wird eingesetzt bei Aktualisierung oder Erzeugung von Informationen zu einer bekannten Entität. Ein PUT wirkt sich direct auf die Ressource aus, deren URI Ziel des Requests ist.²

Systembezogen:

Wenn beispielsweise eine Aktualisierung in einem abonnierten Topic vorgenommen wird oder ein bestimmter Fahrradtyp nach der Aktualisierung zu Verfügung steht.

Ändert ein User sein Profil, wird durch die PUT Methode die Änderung realisiert. Das gleiche gilt für den Fall, wenn Benutzer 1 beispielsweise „Rote Mountainbikes“ abonniert dann sieht er ob das entsprechende Fahrrad zu Verfügung steht.

POST:

Die POST-Methode wird vorwiegend eingesetzt, um dem Server mitzuteilen, dass eine Anforderung des Clients weitere Daten enthält.

POST bietet in diesem Zusammenhang folgende Funktionen an:

- . Übertragung von Nachrichten auf eine Nachrichtengruppe, Mailingliste etc.
- . Mitteilung existierender Ressourcen.³

Systembezogen:

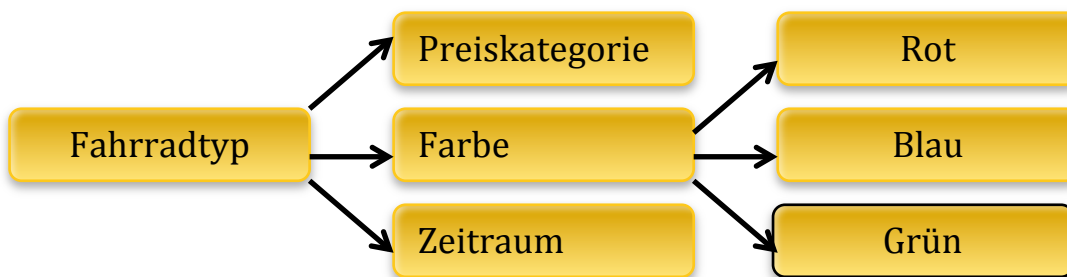
Nach dem Abonnement eines Topics, sowie wenn ein bestimmter Fahrradtyp zu Verfügung steht, erhält der Benutzer eine Nachricht.

² Nach:Stefan Tilkov: Rest for SOA, OOP Conference 2008

³ <http://www.selfphp.info/praxisbuch/praxisbuchseite.php?site=183&group=32>

Topics

Als Topics werden bestimmte Inhalte definiert, die ein Benutzer auswählen kann, z.B:



Fahrradtyp:

- Farbe
- Preiskategorie
- Dauer der Leihe (Wochentarif, Wochenendtarif oder Stundentarif)
„Optional“

Datenstruktur

Benutzerprofil

```
{  
  "title": "benutzerprofil",  
  "type": "object",  
  "properties": {  
    "vorname": { "type": "string" },  
    "nachname": { "type": "string" },  
    "e-mail": { "type": "string" },  
  }  
}
```

Topic//Fahrradtyp

```
{  
  "title" : "fahrradtyp",  
  "type": "object",  
  properties: {  
    "Mountainbike": { "type" : "string"},  
    "Trakkingbike": { "type" : "string"},  
    "Beachcruiser": { "type" : "string"},  
  }  
}
```

Wenn Topic subscriben ist hat der user Zugriff auf Untertopics des entsprechenden Haupttopics:

Mountainbike

```
{  
  "title" : "mountainbike",  
  "type": "object",  
  properties: {  
    "Farbe": { "type" : "string"},  
    "Preis": { "type" : "float"},  
    "Zeitraum": { "type" : "date"},  
  }  
}
```

Abonnieren

```
{  
  "title" : "abonnieren",  
  "type": "object",  
  properties: {  
    "rückmeldung": { "type" : "string"},  
    "verfügbarkeit": { "type" : "boolean"},  
  }  
}
```

Kommunikationsmodell

