



BIKE-SHARING WEBSERVICE MIT CLIENT-APPLIKATION

Tobias Hafermalz, Philipp Thöricht

Dresden, 19.11.2013

Inhalt

Aufgabe

Vorgehen

Funktionen des Webservice

Webclient

Implementierung des OAuth2-Servers

Client-Anwendung

Fazit

Quellen

Aufgabe

- Erstellung eines REST-Webservices
 - Implementierung in PHP mit Slim
 - Authentifizierung durch OAuth2
 - Verschlüsselung durch HTTPS
- Erstellung einer Webanwendung als Client
- Szenario: Bikesharing

Vorgehen

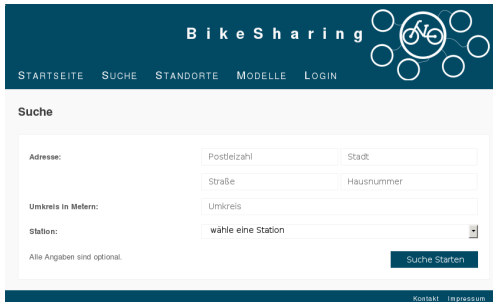
- public API implementiert
- Client (Webanwendung) implementiert
- OAuth2 Server implementiert
- protected APIs implementiert
- Client-Anwendung vervollständigt
- Bugfixes

Funktionen des Webservice

Name	Method	URL	Access
Alle verfügbare Fahrradstationen	GET	/stations	public
Spezielle Station	GET	/stations/stationID	public
Alle verfügbaren Fahrräder	GET	/bikes	public
Spezielles Fahrrad	GET	/bikes/bikesID	public
Alle Fahrradmodelle	GET	/models	public
Spezielles Fahrradmodell	GET	/models/modelID	public
Alle Buchungen	GET	/bookings	protected
Buchung erstellen	POST	/bookings	protected
Einzelne Buchung	GET	/bookings/bookingID	protected
Einzelne Buchung stornieren	DELETE	/bookings/bookingID	protected
Einzelne Buchung bearbeiten	PUT	/bookings/bookingID	protected
Accountinformationen	GET	/account	protected

Webclient

- erst funktionabel, dann schön
- Nutzung von Bibliotheken (jQuery, OpenLayers)
- Gerüst mit PHP erstellt, Inhalt mit Javascript gefüllt
- ansatzweise auch für mobile Endgeräte nutzbar



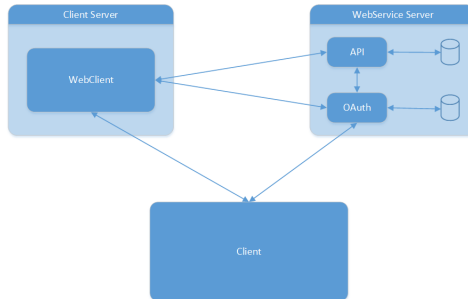
The screenshot shows a web client for a bike-sharing service. The header is dark blue with the text "Bike Sharing" and a logo of a bicycle inside a circle, surrounded by other circles. Below the header is a navigation bar with links: STARTSEITE, SUCHE, STANDORTE, MODELLE, and LOGIN. The main content area is titled "Suche" (Search). It contains a search form with the following fields:

- Adresse:** Postleitzahl (postal code), Stadt (city), Straße (street), and Hausnummer (house number).
- Umkreis in Metern:** Umkreis (radius in meters).
- Station:** wähle eine Station (select a station) with a dropdown arrow.

Below the form, it says "Alle Angaben sind optional." (All data is optional). A blue button labeled "Suche Starten" (Start Search) is at the bottom right of the form. At the very bottom of the page, there are links for "Kontakt" (Contact) and "Impressum" (Imprint).

Implementierung des OAuth2-Servers

- Verwendung der „OAuth2 Server Library for PHP“





Client-Anwendung

Demo

Fazit

- Implementierung des Webclient nach Vorlage einer durchdachten API gut machbar
- Das Slim-Framework war eine gute Wahl, da die Verwendung sehr einfach und fehlerfrei verlief
 - OAuth-Middleware hat leider nicht funktioniert
- Implementierung eines OAuth-Servers ist relativ kompliziert

Quellen

- `slimframework.com`
- `jquery.com`
- `openlayers.org`
- `https://github.com/bshafter/oauth2-server-php`