# SOAP-Endpunkt als REST verfügbar machen.

## Methode mit neuer Annotation ausstatten, dabei UriTemplate auf Methodennamen setzen

[WebInvoke(Method = "POST",RequestFormat = WebMessageFormat.Json,ResponseFormat = WebMessageFormat.Json,BodyStyle =WebMessageBodyStyle.Bare,UriTemplate = "/recalcOffer")]

## Endpunkt für json hinzufügen:

<endpoint address="json" binding="webHttpBinding" bindingConfiguration="myWsHttpBinding" contract="Cic.OpenOne.GateSL.Service.Contract.IslService" behaviorConfiguration="jsonBehaviour" />

## Änderungen zum üblichen Endpunkt

* address -> legt die Adresse für den JSON-Endpunkt fest. Anstatt bisher xy.svc dann xy.svc/json/uriTemplate
* behaviorConfiguration = jsonBehavior
* binding=webHttpBinding

## Endpunkt-Behavior hinzufügen

<behavior name="jsonBehaviour">

<webHttp helpEnabled="true"/>

<crossOriginResourceSharingBehavior />

</behavior>

# REST-Client

Authentication über http-Header übermitteln. Header-Namen sind required und case-sensitive.

|  |  |
| --- | --- |
| UserType | 1 |
| ISOLanguageCode | de-DE |
| Username | OSTERU |
| Password | XAKLOP901ASDDDA |
| SysPEROLE | 76 |

Jersey-JAX-RS

* Header wie oben beschrieben setzen
* als Post-Entity kann eine beliebige Instanz verwendet werden

Client client = ClientBuilder.newClient();

WebTarget target = client.target("http://localhost:60272/slService.svc/json/recalculateOffer");

String inputData = "22283";

Response resp = target.request(MediaType.APPLICATION\_JSON)

.header("Username","OSTERU")

.header("Password","XAKLOP901ASDDDA")

.header("SysPEROLE","76")

.header("ISOLanguageCode","de-DE")

.header("UserType","1")

.post(Entity.json(inputData));

//unmarshall to response-Object

OBaseDto responseTest = resp.readEntity(OBaseDto.class);