



# Verteilte Verarbeitung

Kapitel 12.3

Dokumentation einer Schnittstelle  
mit Swagger

# Was ist Swagger?

- Dokumentation für WebServices mit SOAP = WSDL
- Variante für RESTful WebServices (WADL) hat sich nicht durchgesetzt
- Daher drei Alternativen
  - Swagger (hier erläutert)
  - RAML
  - API Blueprint
- Swagger (<http://swagger.io/>) sehr weit verbreitet
- Integration in viele Programmiersprachen und APIs (auch Java)
- Dokumentation jeder URI mit samt der HTTP-Verben und HTTP Return Codes
  - Über spezielles Java API dokumentierbar
  - GUI in Java Script verfügbar (Swagger UI)

# Notwendige Voraussetzungen

## build.gradle

```
dependencies {  
    ...  
    compile 'io.springfox:springfox-swagger2:2.9.2'  
    compile 'io.springfox:springfox-swagger-ui:2.9.2'  
    ...  
}
```

Sowie:

```
@Configuration  
@EnableSwagger2  
public class SwaggerConfig {  
    @Bean  
    public Docket api() {  
        return new Docket(DocumentationType.SWAGGER_2)  
            .select()  
            .apis(RequestHandlerSelectors.any())  
            .paths(PathSelectors.any())  
            .build();  
    }  
}
```

# Alternative zur Beschreibung: Swagger

**swagger**   **Explore**

**kunden** [Show/Hide](#) [List Operations](#) [Expand Operations](#)

GET	/kunden	<a href="#">Kunden Liste</a>
POST	/kunden	
DELETE	/kunden/{nummer}	
GET	/kunden/{nummer}	
PUT	/kunden/{nummer}	

[ BASE URL: / , API VERSION: 1.0.2 ]

# Dokumentation einer Methode

**kunde-controller** Kunde Controller

**GET** /api/v1/kunden/{nummer} Sucht einen Kunden mit der uebergebenen Kundennummer

Bisher nur als Dummy

**Parameters** [Try it out](#)

Name	Description
<b>nummer</b> <small>* required</small>	nummer
string	
(path)	

**Responses** Response content type application/json

**Curl**

```
curl -X GET "http://localhost:8080/api/v1/kunden/123456" -H "accept: application/json"
```

**Request URL**

```
http://localhost:8080/api/v1/kunden/123456
```

**Responses**

Code	Description
200	Kunde gefunden
	<small>Example Value Model</small>
	<pre>{  "nachname": "string",  "nummer": 0,  "vorname": "string"}</pre>
400	Ungueltige Id
401	Unauthorized
403	Forbidden
404	Kunde nicht gefunden

# Dokumentation mit Swagger im Quelltext

```
@RequestMapping(  
    value="kunden/{nummer}",  
    method = RequestMethod.GET,  
    produces = MediaType.APPLICATION_JSON_VALUE  
)  
  
@ApiOperation( // SWAGGER  
    value = "Sucht einen Kunden mit ...",  
    notes = "Bisher nur als Dummy",  
    response = Kunde.class,  
    produces = "application/json")  
  
@ApiResponses(value = { // SWAGGER  
    @ApiResponse(code = 200, message = "Kunde gefunden"),  
    @ApiResponse(code = 400, message = "Ungueltig"),  
    @ApiResponse(code = 404, message = "Nicht gefunden") })  
  
public ResponseEntity<Kunde>  
    findKunde(@PathVariable("nummer") String nummer) { ...
```

# RESTful Interface liefert JSON-Datei als Beschreibung des Interfaces

swagger.json:

```
{ "swagger": "2.0",  
  "info": { "version": "1.0.2", "basePath": "/" },  
  "tags": [ { "name": "kunden" } ],  
  "paths": {  
    "/kunden": {  
      "get": {  
        "tags": [ "kunden" ],  
        "summary": "Kunden Liste",  
        "description": "Liefert alle Kunden als Liste",  
        "operationId": "kundenListe",  
        "produces": [ "application/xml", "application/json" ],  
        "parameters": [],  
        "responses": { "200": { "description": "OK" },  
                       "500": { "description": "Something wrong" } } }  
    }  
  }
```