

OpenAPI



Präsentiert von

Furkan Öztürk & Alexander Beckerbauer



Themen

1. Was ist OpenAPI?
2. Wo wird OpenAPI verwendet? / Einsatzgebiete
3. Vorteile von OpenAPI
4. Beispiele



Was ist OpenAPI?

- OpenAPI ist ein Standard/Spezifikation zur Beschreibung von APIs (Programmierschnittstellen engl. Application Programming Interfaces), mit dem sich REST-konforme APIs beschreiben lassen
- Dies bedeutet, es ist ein Dokument, welches beschreibt, was eine auf REST-basierende API genau macht und andere Informationen über diese API
- Das Open-API Dokument wird im YAML oder JSON-Format generiert
- z.B. ein Programmierer, welcher neu zu einem Projekt dazustößt kann sich so schnell einen Überblick über die Funktionen (Request und Response Formate) und andere Spezifikationen der implementierten APIs machen



Einsatzgebiete

- OpenAPI wird verwendet um auf REST basierende APIs auf einheitliche Weise zu beschreiben
- Diese API-Dokumentation wird generiert und ist im maschinenlesbaren Format verfügbar, zudem ist es für Entwickler ebenfalls einfach lesbar
- Bei Änderungen in dem Programm, ändert sich die API-Dokumentation ebenfalls automatisch und eine neue Dokumentation wird generiert



Vorteile von OpenAPI

- Syntax-Tests und Funktionstests sind in OpenAPI integriert
- Die Dokumentation ist sowohl von Menschen als auch von Maschinen lesbar
- Zeitersparnis dadurch, dass die OpenAPI Dokumentation automatisch erzeugt wird und nicht manuell geschrieben werden muss
- Sicherheitslücken können erkannt werden
- Viele Verschiedene Tools mit denen man arbeiten kann



Beispiel

So sieht ein Beispiel einer
Open-API Spezifikation aus

```
openapi: "3.0.0"
info:
  version: 1.0.0
  title: Swagger Petstore
  license:
    name: MIT
servers:
  - url: http://petstore.swagger.io/v1
paths:
  /pets:
    get:
      summary: List all pets
      operationId: listPets
      tags:
        - pets
      parameters:
        - name: limit
          in: query
          description: How many items to return at one time (max 100)
          required: false
          schema:
            type: integer
            format: int32
      responses:
        '200':
          description: A paged array of pets
          headers:
            x-next:
              description: A link to the next page of responses
              schema:
                type: string
          content:
            application/json:
              schema:
                $ref: "#/components/schemas/Pets"
```

[3]
<https://support.smartbear.com/swaggerhub/docs/tutorials/openapi-3-tutorial.html>



Meta Info

```
openapi: 3.0.0
```

```
info:
```

```
  version: 1.0.0
```

```
  title: Simple Artist API
```

```
  description: A simple API to illustrate  
  OpenAPI concepts
```

Paths (Endpoint URL)



```
servers:
  - url: https://example.io/v1
components:
  securitySchemes:
    BasicAuth:
      type: http
      scheme: basic
  security:
    - BasicAuth: []
paths:
  /artists:
    get:
      description: Returns a list of artists
```


Responses



```
responses:
  '200':
    description: Successfully returned a list of artists
    content:
      application/json:
        schema:
          type: array
          items:
            type: object
            required:
              - username
            properties:
              artist_name:
                type: string
              artist_genre:
                type: string
              albums_recorded:
                type: integer
              username:
                type: string
```



Weitere Teile der Open API-Spezifikation

- Query-Parameter
- POST, PUT und PATCH Requests -> werden verwendet wenn etwas verändert werden soll
- Parameter für Paths -> Werden verwendet, wenn nur spezifische Anfragen passieren sollen