Escuela de Ingeniería Informática

Universidad de Oviedo

SOFTWARE ARCHITECTURE

Laboratory 9

API Description Languages - OpenAPI

2025-26

Pablo González
Jose Emilio Labra Gayo
Diego Martín
Celia Melendi

School of Computer Science, University of Oviedo

APIs Description – What is it?

- Human-Machine Readable Specification of a RESTful API.
- Facilitates its use and comprehension.
- Enables automation of:
    - Creation of Documentation
    - Code SDKs
    - Creation of Test Cases
    - Mocking and Stubbing

APIs description languages

- Web Services Description Language (WSDL)
  - Commonly used in SOAP (XML)
- **OpenAPI (REST)**

OpenAPI - What is it?

OPENAPI INITIATIVE

- Description of HTTP-based APIs (commonly RESTfull) that eases its usage and comprehension by the consumer (user-application).
- Comprised of a YAML file (API specification) that delineates inputs, outputs, and guidelines for accessing and consuming it.
- This specification can be manually created or automatically generated using a tool directly from the code itself.

OpenAPI
API Specification

- A self-contained resource that defines the elements of the API.
  - It must include at least:
    - One `path` field.
    - One `components` or `webhook` field.
- In the specification, we can find:
  - Path Schemes: These specify different parameters (e.g., the number of questions in `/questionservice/{}`)
  - Resource Types: Including text-plain, JSON, text + JSON, html + JSON…
  - HTTP Status Codes: Used to indicate results, such as 500 for errors or 200 for success

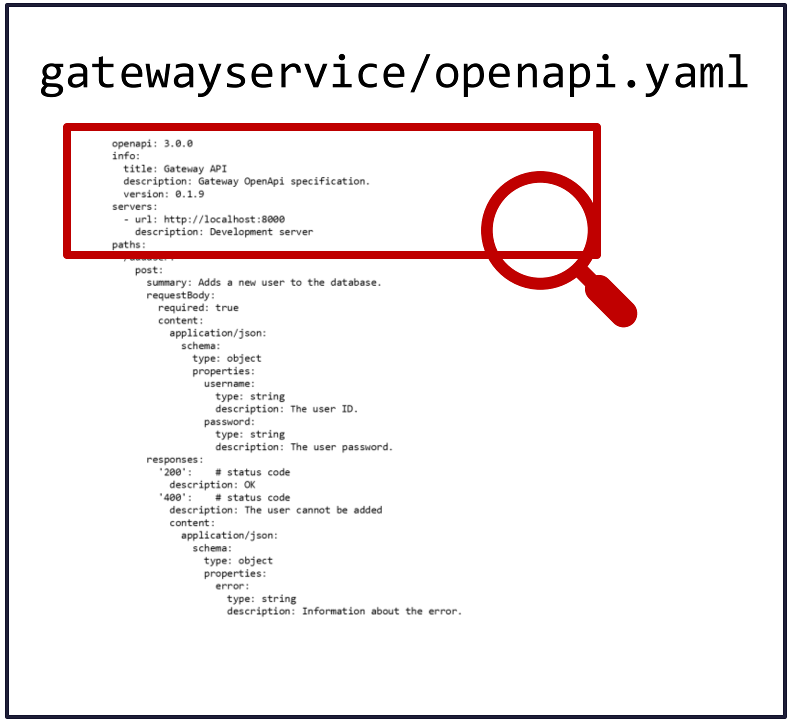
API Specification- Example

```yaml
openapi: 3.0.0
info:
  title: Gateway API
  description: Gateway OpenApi specification.
  version: 0.1.9
servers:
  - url: http://localhost:8000
    description: Development server
paths:
  …
```

OpenAPI version

Information: name, description and API version

Servers-Host on which the API is made available



gatewayservice/openapi.yaml

```yaml
openapi: 3.0.0
info:
  title: Gateway API
  description: Gateway OpenApi specification.
  version: 0.1.9
servers:
  - url: http://localhost:8000
    description: Development server
paths:
  /adduser:
    post:
      summary: Adds a new user to the database.
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                username:
                  type: string
                  description: The user ID.
                password:
                  type: string
                  description: The user password.
      responses:
        '200':    # status code
          description: OK
        '400':    # status code
          description: The user cannot be added
          content:
            application/json:
              schema:
                type: object
                properties:
                  error:
                    type: string
                    description: Information about the error.
```

API Specification- Example

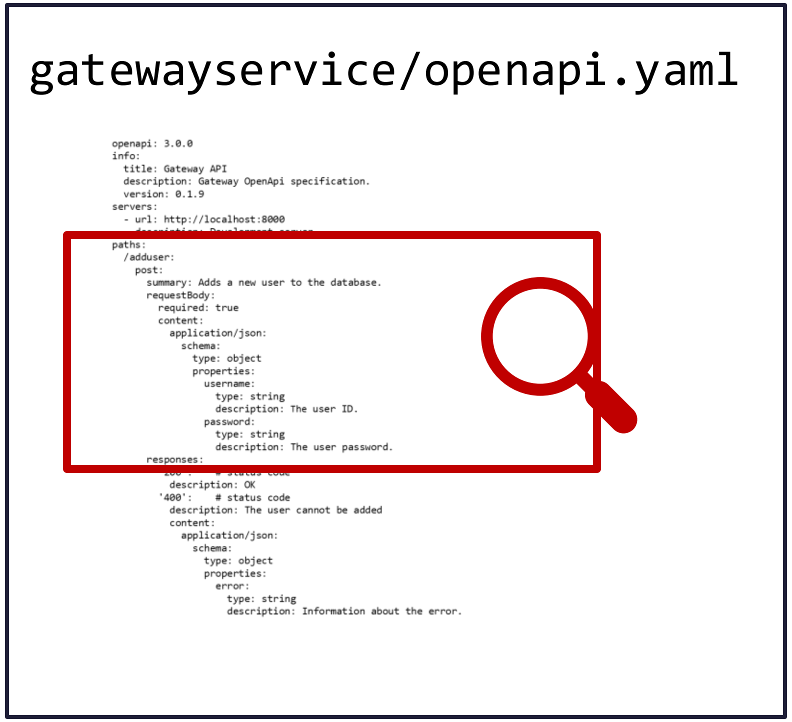
```
…
paths:
  /adduser:
    post:
      summary: Adds a new user to the database.
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                username:
                  type: string
                  description: The user ID.
                password:
                  type: string
                  description: The user password.
      responses:
…
```

Paths: one for each available endpoint

Type of operation

Request body specification: (if is mandatory): content, type and scheeme

Type and description of each attribute



gatewayservice/openapi.yaml

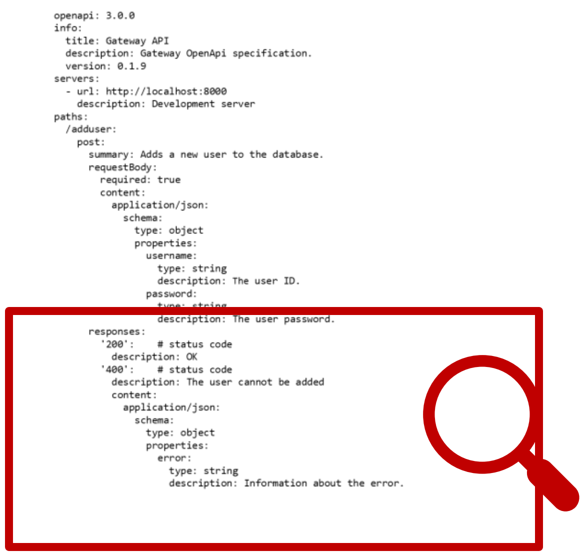
```
openapi: 3.0.0
info:
  title: Gateway API
  description: Gateway OpenApi specification.
  version: 0.1.9
servers:
  - url: http://localhost:8000
    description: Development server
paths:
  /adduser:
    post:
      summary: Adds a new user to the database.
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                username:
                  type: string
                  description: The user ID.
                password:
                  type: string
                  description: The user password.
      responses:
        '200':    # status code
          description: OK
        '400':    # status code
          description: The user cannot be added
          content:
            application/json:
              schema:
                type: object
                properties:
                  error:
                    type: string
                    description: Information about the error.
```

API Specification- Example

```
…
responses:
   '200':    # status code
      description: OK
   '400':    # status code
      description: The user cannot be added
      content:
        application/json:
          schema:
            type: object
            properties:
              error:
                type: string
                description: Information about the error.
```

Path outputs (responses): descriptions, types, and content with their schemes for each

gatewayservice/openapi.yaml



```
openapi: 3.0.0
info:
  title: Gateway API
  description: Gateway OpenApi specification.
  version: 0.1.9
servers:
  - url: http://localhost:8000
    description: Development server
paths:
  /adduser:
    post:
      summary: Adds a new user to the database.
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                username:
                  type: string
                  description: The user ID.
                password:
                  type: string
                  description: The user password.
      responses:
        '200':    # status code
          description: OK
        '400':    # status code
          description: The user cannot be added
          content:
            application/json:
              schema:
                type: object
                properties:
                  error:
                    type: string
                    description: Information about the error.
```

Good Practices

OPENAPI INITIATIVE

- Defining the specification before its implementation:
  - Enables us to focus on functionality and identify issues earlier.
  - Ease interoperability between different types of services.

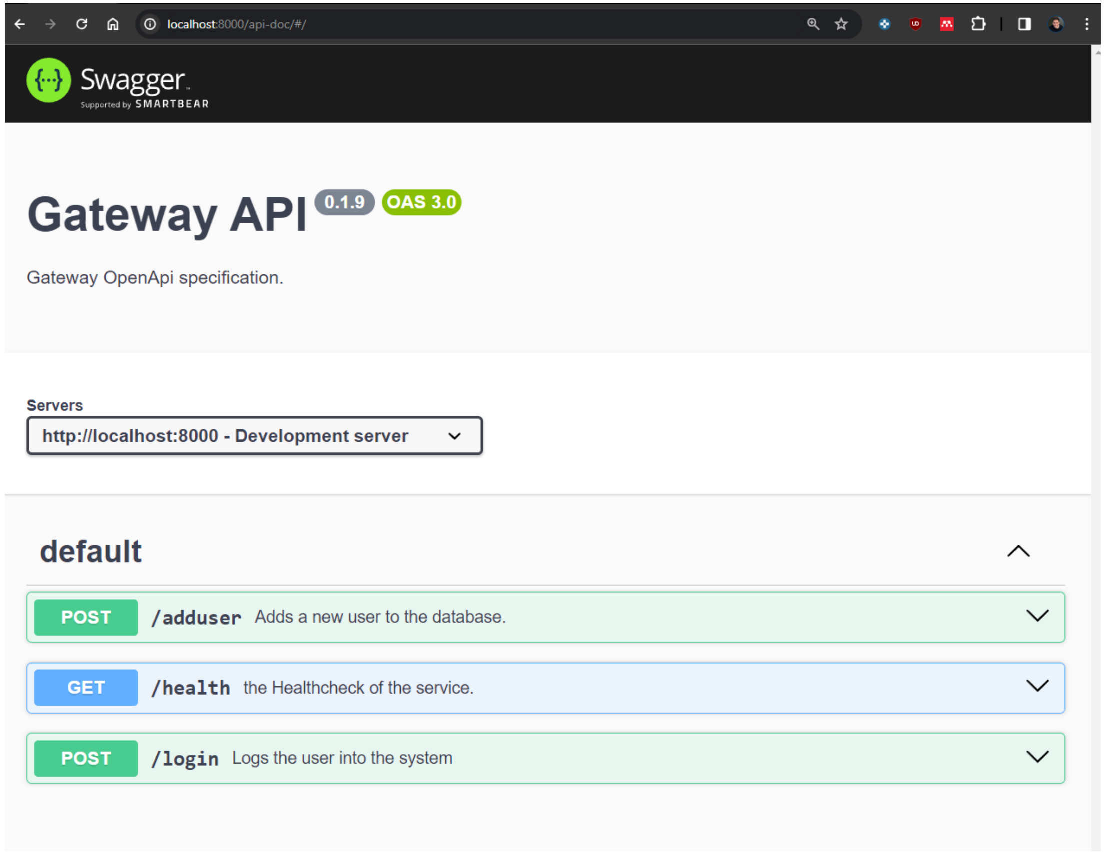
- Synchronize/update the specification with the API changes [1]

[1] Alberto Martin-Lopez, Sergio Segura, and Antonio Ruiz-Cortés. FSE22. Online testing of RESTful APIs: promises and challenges.

School of Computer Science, University of Oviedo

Swagger

- Toolkit that eases the API specification deployment
- Enables:
  - Read the specification in a easy and clear way.
  - Check the different endpoints and hosts where the API is made available.
  - Available in the Gateway service (gatewayservice) of wichat_0:
  - http://localhost:8000/api-doc



localhost:8000/api-doc/#/

Swagger
Supported by SMARTBEAR

Gateway API 0.1.9 OAS 3.0

Gateway OpenApi specification.

Servers

http://localhost:8000 - Development server

default

- POST `/adduser` Adds a new user to the database.
- GET `/health` the Healthcheck of the service.
- POST `/login` Logs the user into the system