



សាកលវិទ្យាល័យ ពុទ្ធិសាស្ត្រ
UNIVERSITY OF PUTHISAstra

ASSESSMENT I - Swagger

Lecturer: Sok Piseth

Presented by: Theng Thon

10th January 2025



CONTENT

1. Introduction to Swagger
2. Importance of API Documentation
3. Core Components of Swagger
4. Benefits of Using Swagger
5. How to use Swagger?
6. Swagger vs. Postman
7. Tools & Resources
8. Conclusion
9. Demonstration

01. Introduction to Swagger

What is Swagger?

- Swagger is an open-source framework used for designing, building, documenting, and consuming RESTful APIs.
- It simplifies the process of documenting APIs and enables easy integration between services.



01. Introduction to Swagger

History of Swagger:

- Originally created by Tony Tam in 2010.
- Later acquired by SmartBear, it evolved into OpenAPI Specification (OAS), but the name "Swagger" is still widely used.



02. Importance of API Documentation

Why API Documentation is Necessary?

- Helps developers and teams understand the functionality and usage of the API.
- Improves collaboration between frontend, backend, and third-party developers.
- Ensures better maintainability of APIs over time.



02. Importance of API Documentation

Challenges Without Documentation:

- Harder for new developers to understand and use the API.
- Increased risk of bugs and miscommunication between services.



03. Core Components of Swagger

1. Swagger Specification (OAS): Describes the API, including its paths, parameters, and responses in a machine-readable format (YAML/JSON).

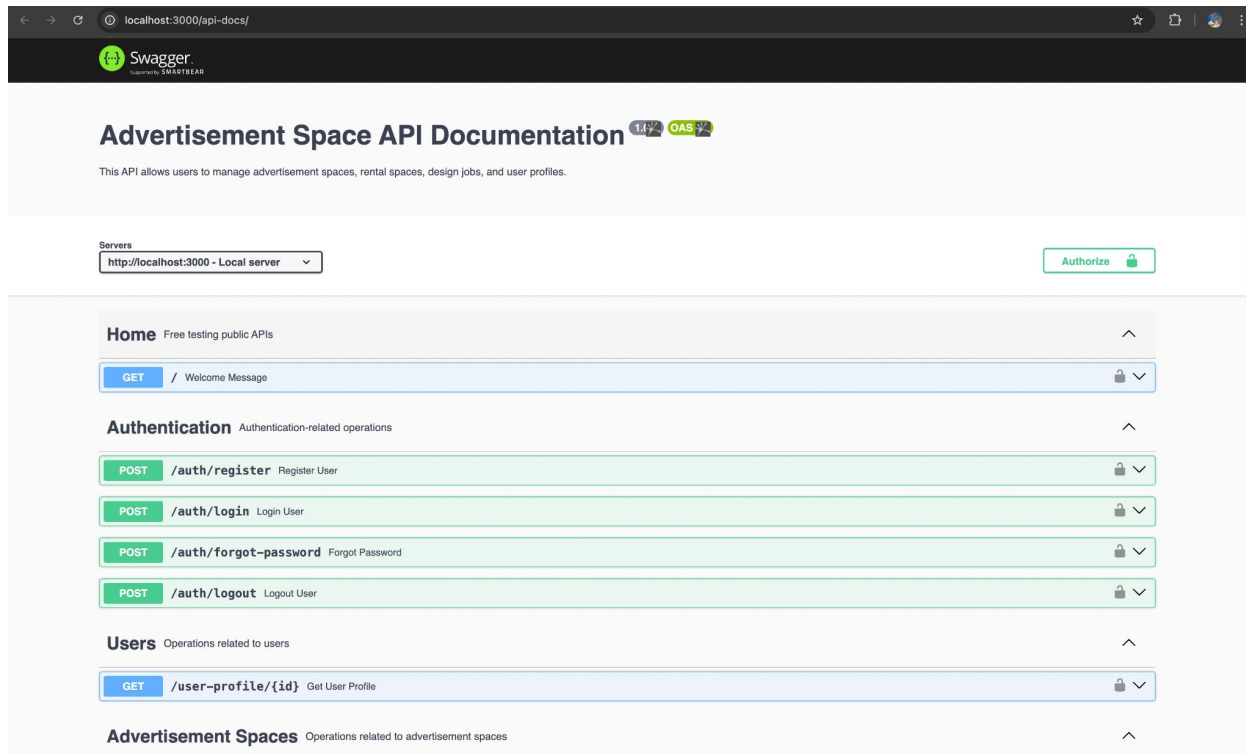
yml

```
openapi: 3.0.0
info:
  title: Sample API
  version: 1.0.0
  description: A simple API example
paths:
  /users:
    get:
      summary: Retrieve users
      responses:
        '200':
          description: A list of users
```

```
const swaggerConfig = {
  openapi: '3.0.0',
  security: [
    {
      BearerAuth: [],
    },
  ],
  info: {
    title: 'Advertisement Space API Documentation',
    version: '1.0.0',
    description:
      'This API allows users to manage advertisement spaces',
  },
  servers: [
    {
      url: 'http://localhost:3000',
      description: 'Local server',
    },
  ],
  tags: [...],
  components: {...},
  paths: {...},
};
```

03. Core Components of Swagger

2. Swagger UI: A graphical interface that automatically generates and displays API documentation, allowing developers to interact with the API directly.



03. Core Components of Swagger

3. Swagger Editor: An editor for creating and validating OpenAPI specifications.

The screenshot displays the Swagger Editor web application. The left sidebar shows the Swagger Editor logo and navigation links: File, Edit, Insert, Generate Server, Generate Client, and About. The main editor area displays the Swagger Petstore - OpenAPI 3.0 specification in YAML format. The right sidebar shows the rendered Swagger UI for the Petstore API, including the title, version (1.0.11), and a list of endpoints with their methods and descriptions.

```
1 openapi: 3.0.3
2 info:
3   title: Swagger Petstore - OpenAPI 3.0
4   description: |-
5     This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can
6     find out more about
7     Swagger at [https://swagger.io](https://swagger.io). In the third iteration of the
8     pet store, we've switched to the design first approach!
9     You can now help us improve the API whether it's by making changes to the definition
10    itself or to the code.
11    That way, with time, we can improve the API in general, and expose some of the new
12    features in OAS3.
13
14    _If you're looking for the Swagger 2.0/OAS 2.0 version of Petstore, then click
15    [here](https://editor.swagger.io/?url=https://petstore.swagger.io/v2/swagger
16    .yaml). Alternatively, you can load via the 'Edit > Load Petstore OAS 2.0' menu
17    option!_
18
19    Some useful links:
20    - [The Pet Store repository](https://github.com/swagger-api/swagger-petstore)
21    - [The source API definition for the Pet Store](https://github.com/swagger-api
22    /swagger-petstore/blob/master/src/main/resources/openapi.yaml)
23
24  termsOfService: http://swagger.io/terms/
25  contact:
26    email: apiteam@swagger.io
27  license:
28    name: Apache 2.0
29    url: http://www.apache.org/licenses/LICENSE-2.0.html
30  version: 1.0.11
31  externalDocs:
32    description: Find out more about Swagger
33    url: http://swagger.io
34  servers:
35    - url: https://petstore3.swagger.io/api/v3
36  tags:
37    - name: pet
38      description: Everything about your Pets
39      externalDocs:
40        description: Find out more
41        url: http://swagger.io
42    - name: store
43      description: Access to Petstore orders
44      externalDocs:
45        description: Find out more about our store
46        url: http://swagger.io
47    - name: user
48      description: Operations about user
49  paths:
50    /pet:
```

Swagger Petstore - OpenAPI 3.0 1.0.11 OAS 3.0

This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can find out more about Swagger at <https://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help us improve the API whether it's by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS3.

If you're looking for the Swagger 2.0/OAS 2.0 version of Petstore, then click [here](#). Alternatively, you can load via the [Edit > Load Petstore OAS 2.0](#) menu option!

Some useful links:

- [The Pet Store repository](#)
- [The source API definition for the Pet Store](#)

Terms of service
Contact the developer
Apache 2.0
Find out more about Swagger

Servers
 [Authorize](#)

pet Everything about your Pets [Find out more](#)

PUT /pet Update an existing pet

POST /pet Add a new pet to the store

GET /pet/findByStatus Finds Pets by status


GET /pet/findByTags Finds Pets by tags

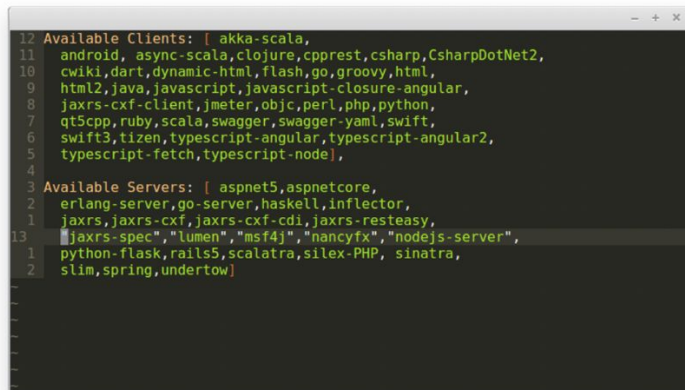
03. Core Components of Swagger

4. Swagger Codegen: Generates client libraries, server stubs, and API documentation based on the OpenAPI specification.

Swagger Codegen

Swagger Codegen can simplify your build process by generating server stubs and client SDKs for any API, defined with the OpenAPI (formerly known as Swagger) specification, so your team can focus better on your API's implementation and adoption.

 Download



```
12 Available Clients: [ akka-scala,
11 android, async-scala, clojure, cpprest, csharp, CsharpDotNet2,
10 cwiki, dart, dynamic-html, flash, go, groovy, html,
9 html2, java, javascript, javascript-closure-angular,
8 jaxrs-cxf-client, jmeter, objc, perl, php, python,
7 qt5cpp, ruby, scala, swagger, swagger-yaml, swift,
6 swift3, tizen, typescript-angular, typescript-angular2,
5 typescript-fetch, typescript-node],
4
3 Available Servers: [ aspNet5, aspnetcore,
2 erlang-server, go-server, haskell, inflector,
1 jaxrs, jaxrs-cxf, jaxrs-cxf-cdi, jaxrs-resteasy,
13 jaxrs-spec, "lumen", "msf4j", "nancyfx", "nodejs-server",
1 python-flask, rails5, scalatra, silex-PHP, sinatra,
2 slim, spring, undertow]
```

04. Benefits of Using Swagger

- > **Interactive Documentation:** Swagger UI provides an interactive interface to test API endpoints directly from the documentation.
- > **Standardization:** Swagger uses OpenAPI, a widely adopted standard for API documentation.
- > **Automation:** Automatically generates API documentation from the Swagger specification.
- > **Code Generation:** Tools like Swagger Codegen enable the automatic creation of client SDKs and server stubs.
- > **Consistency:** By adhering to the OpenAPI Specification, teams ensure a consistent approach to API design and documentation.

05. How to use Swagger?

- > **Step 1:** Install Swagger tools (swagger-ui-express, swagger-jsdoc).
- > **Step 2:** Write your API documentation in OpenAPI specification format (YAML/JSON).
- > **Step 3:** Use Swagger UI to display the API documentation interactively.
- > **Step 4:** Optionally, use Swagger Codegen to generate client libraries and server stubs.

06. Swagger Vs Postman



06. Swagger Vs Postman

Feature	Swagger	Postman
Primary Use Case	API Design, Documentation, and Specification	API Testing, Exploration, and Automation
Format	OpenAPI Specification (OAS), YAML/JSON	Custom Postman Collections
Interactive Documentation	Yes, Swagger UI provides a rich interactive UI for testing endpoints.	Yes, Postman allows interaction with APIs, but not for documentation directly.
Testing Capabilities	Limited to API documentation and testing basic endpoints in Swagger UI.	Extensive testing features, including automated tests with assertions, environments, and scripting.
Client SDK Generation	Yes, using Swagger Codegen (supports multiple languages).	No, Postman doesn't offer SDK generation.

07. Tools & Resources

- > **Swagger UI:** <https://petstore.swagger.io/>
- > **Swagger Editor:** <https://editor.swagger.io/>
- > **Swagger Codegen:** <https://swagger.io/tools/swagger-codegen/>
- > **Swagger Specifications / Open API Specification (OAS):**
<https://swagger.io/specification/>

08. Conclusion

> Key Takeaways:

- Swagger is a comprehensive suite of tools for designing, documenting, and testing APIs.
- OpenAPI (Swagger) provides a standard for API documentation that is widely accepted in the industry.
- Swagger UI and Codegen are invaluable tools for developers working with APIs.

> Next Steps:

- Start using Swagger in your projects for better API design and documentation.
- Experiment with Swagger Codegen to generate SDKs for your APIs.

09. Demonstration



Thank You