

**TAKE  
THE  
LEAD**

# Unlocking the Power of Spec-Driven API Development

Antwerpen

...:::•

Sep 2024



I'm **Tim Haselaars**

**Product Manager at**



API & Event enthusiast

**Opensource contributor**

on [Portman](#), [Postman-to-K6](#), [OpenAPI-format](#), [EventCatalog](#)



**OPENAPI-FORMAT**



**EventCatalog**



# Table of contents

01

**Kick-off**

02

**Specifications**

03

**Spec Driven Development**

04

**Workflow examples**

# MARVEL

REST API

Congrats!  
You just launched your API 



Where can I find the docs?



# How fast can she go?



# How can I play with it?



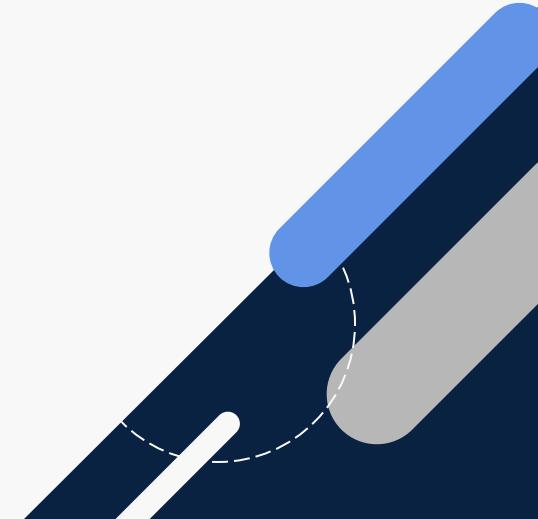
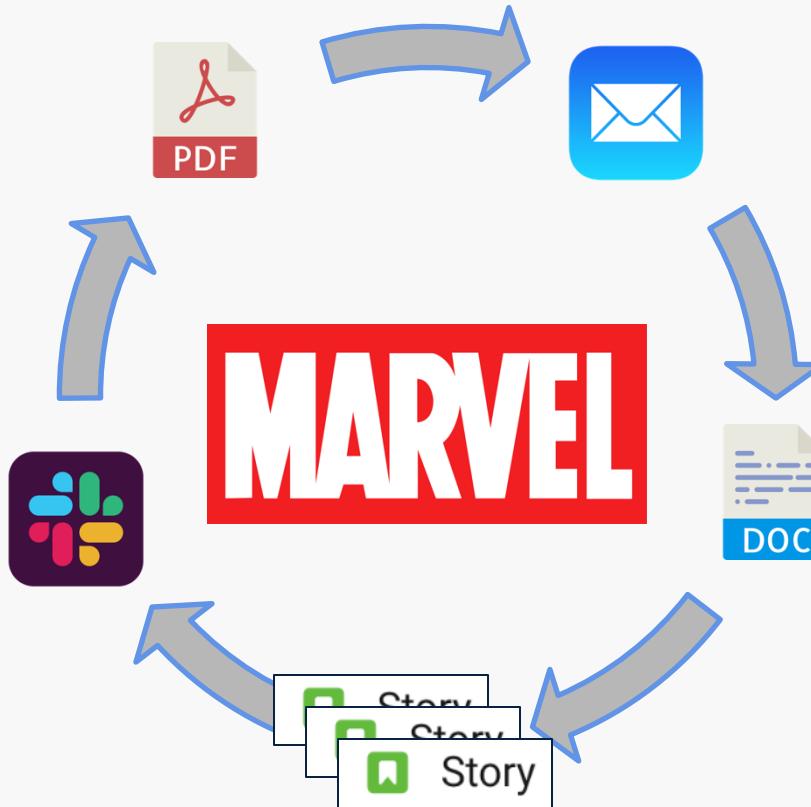
# Which fields are required?



You got a feature request  
Start all over again!



Let's quickly create, some ...



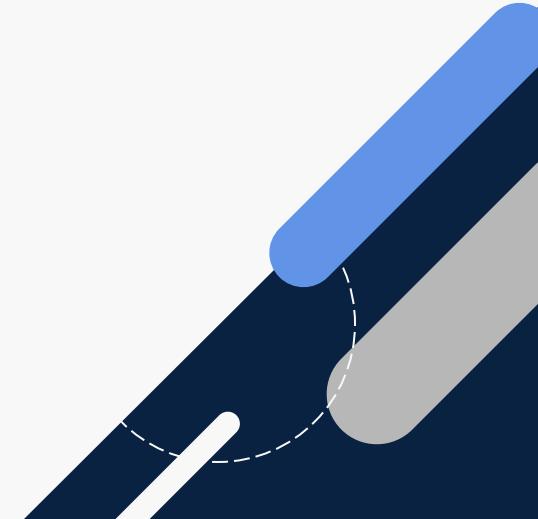
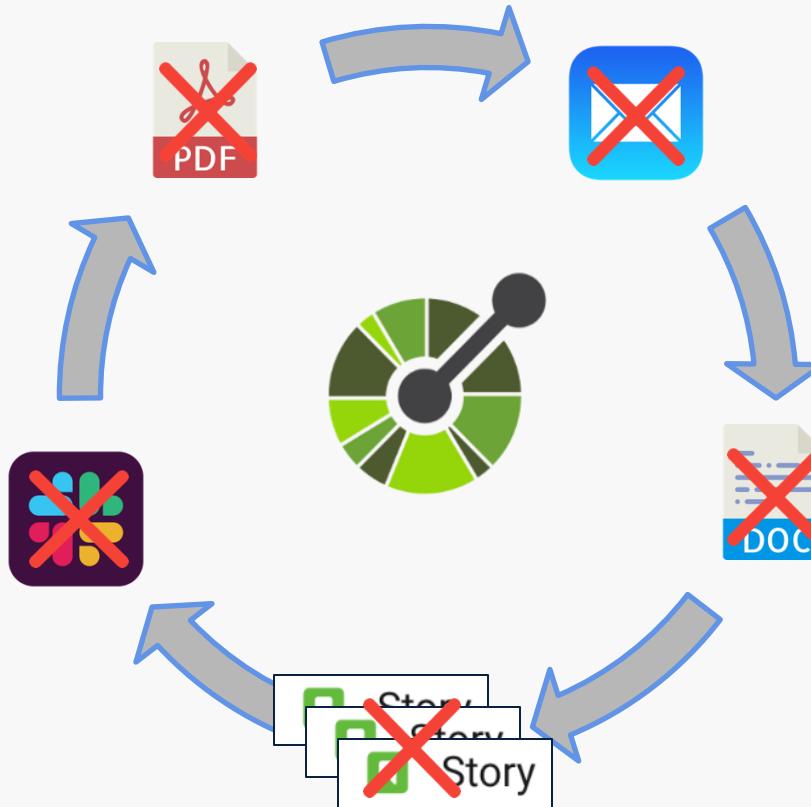
You are not alone  
API docs are in poor condition

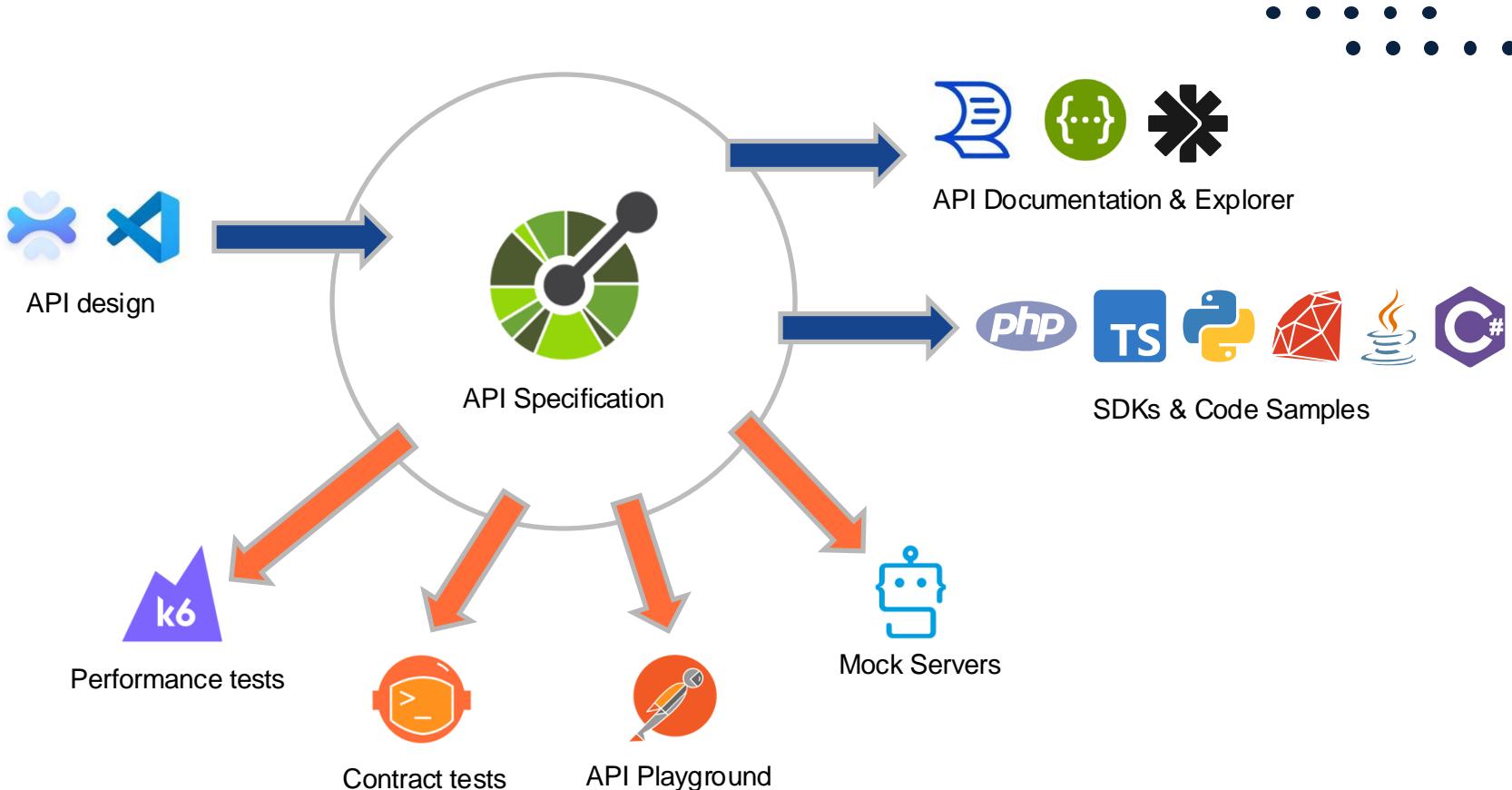


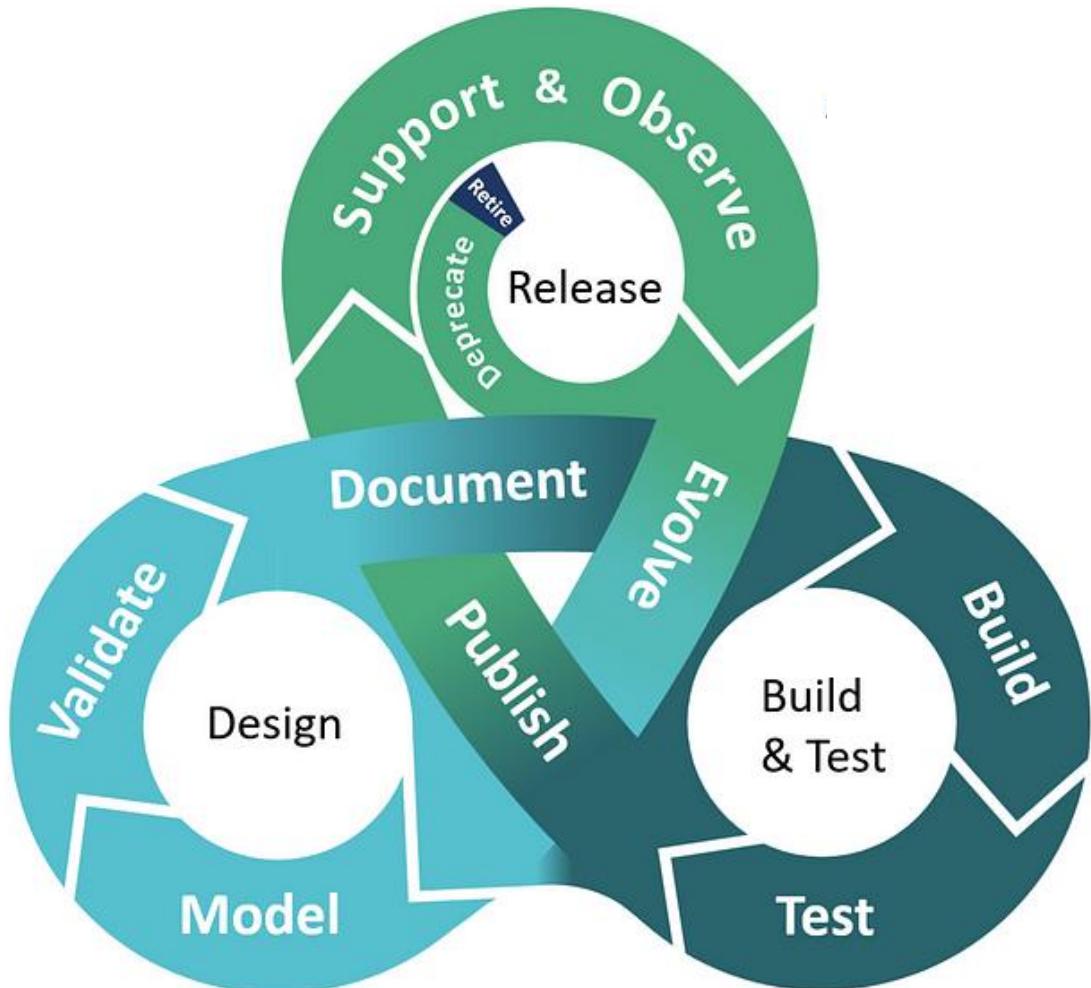
Let's fix this



# Let's stop, doing one-off efforts







# Single source of truth

## API specification

- **Describes** general information about the API
- Available **paths** e.g. /resources
- Available **operations** on each path e.g. GET /resources/fid{}
- **input & output** for each operation
- Model **schemas**, describing the fields/properties
- **Security & Authentication**



api blueprint

WADL

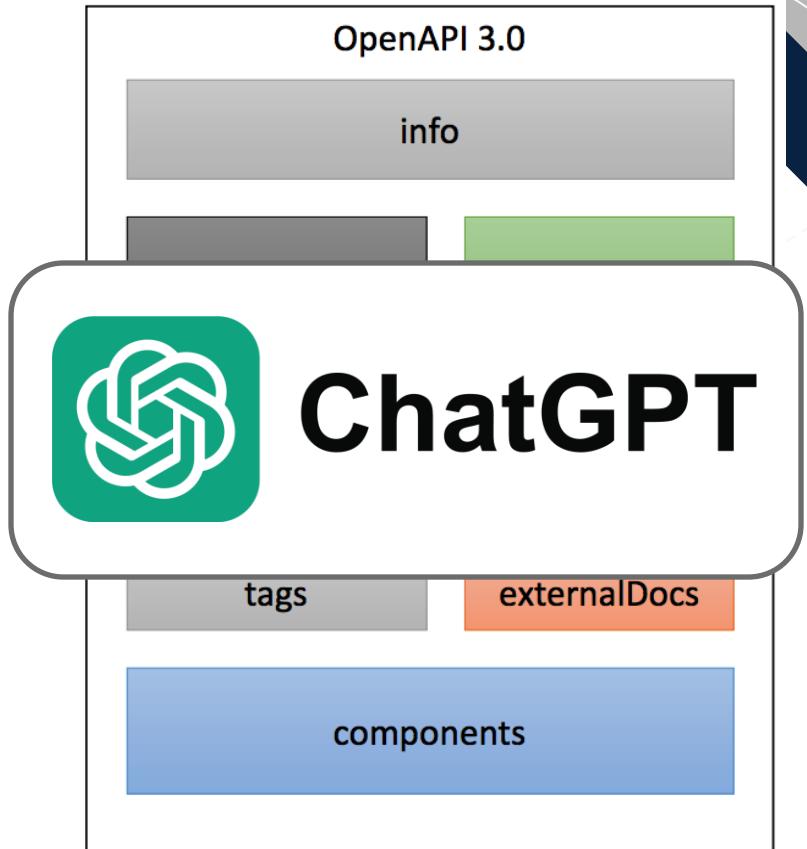
• • • • •



# OPENAPI

- Worldwide Standard
- Formerly named Swagger (2.0)- 2011
- Current version
  - 3.0 - 2017
  - 3.1 - 2021
- Format: JSON or YAML
- REST API
- JSON Schema
- Independent of language/framework

• • • • •





## Ecosystem

- Open-source & Commercial tooling
- Supported by big brands



Microsoft

Google



IBM

• • • •

The screenshot shows a web browser window with the title "OpenAPI.Tools - an Open Source Project". The address bar contains "openapi.tools". The main content area has a purple header with the text "OpenAPI.Tools" in white. Below the header, a sub-header reads "Stay up to date with a community-driven list of high-quality, modern tools for OpenAPI. An open source project from APIs You Won't Hate." A message encourages users to submit missing tools via GitHub. A green button labeled "Submit an OpenAPI tool" is visible.

# OpenAPI.Tools

Stay up to date with a community-driven list of high-quality, modern tools for OpenAPI. An open source project from [APIs You Won't Hate](#).

Is there a tool missing that you think should be on the list?  
Submit a Pull Request on GitHub [apisyouwonthate/openapi.tools](#)

[Submit an OpenAPI tool](#)

## Tool Types

We've organised everything into categories so you can jump to the section you're interested in.

- **Auto Generators:** Tools that will take your code and turn it into an OpenAPI Specification document
- **Converters:** Various tools to convert to and from OpenAPI and other API description formats.
- **Data Validators:** Check to see if API requests and responses are lining up with the API description.
- **Description Validators:** Check your API description to see if it is valid OpenAPI.
- **Documentation:** Render API Description as HTML (or maybe a PDF) so slightly less technical people can figure out how to work with the API.
- **DSL:** Writing YAML by hand is no fun, and maybe you don't want a GUI, so use a Domain Specific Language to write OpenAPI in your language of choice.
- **Gateways:** API Gateways and related tools that have integrated support for OpenAPI.
- **GUI Editors:** Visual editors help you design APIs without needing to memorize the entire OpenAPI specification.
- **Learning:** Whether you're trying to get documentation for a third party API based on traffic, or are trying to switch to design-first at an organization with no OpenAPI at all, learning can help you move your API spec forward and keep it up to date.
- **Miscellaneous:** Anything else that does stuff with OpenAPI but hasn't quite got enough to warrant its own category.
- **Mock Servers:** Fake servers that take description document as input, then route incoming HTTP requests to example responses or dynamically generates examples.
- **Monitoring:** Monitoring tools let you know what is going on in your API.
- **Parsers:** Loads and read OpenAPI descriptions, so you can work with them programmatically.
- **SDK Generators:** Generate code to give to consumers, to help them avoid interacting at a HTTP level.
- **Security:** By poking around your OpenAPI description, some tools can look out for attack vectors you might not have noticed.
- **Server Implementations:** Easily create and implement resources and routes for your APIs.
- **Testing:** Quickly execute API requests and validate responses on the fly through command line or GUI interfaces.

Repository search results +

github.com/search?q=openapi&type=repositories

Filter by

- <> Code 7.7M
- Repositories 20.1k
- Issues 182k
- Pull requests 491k
- Discussions 5k
- Users 197
- More

Languages

- Java
- JavaScript
- TypeScript
- Python
- Go
- C#
- PHP
- HTML
- Kotlin
- Shell
- More languages...

Advanced

- Owner
- Size
- Number of followers
- Number of forks
- Number of stars
- Only curated

20.1k results (324 ms)

Sort by: Best match Save ...

Single sign-on to see results in the apideck-io organization.

 **OAI/OpenAPI-Specification**  
The OpenAPI Specification Repository  
[rest](#) [openapi](#) [apis](#) [oas](#) [webapi](#)  
JavaScript · ⭐ 27.7k · Updated 6 hours ago

 **OpenAPITools/openapi-generator**  
OpenAPI Generator allows generation of API client libraries (SDK generation), server stubs, documentation and configuration automatically...  
[api](#) [sdk](#) [rest](#) [generator](#) [rest-api](#)  
Java · ⭐ 18.8k · Updated 1 hour ago

 **Redocly/redoc**  
OpenAPI/Swagger-generated API Reference Documentation  
[reactjs](#) [swagger](#) [api-documentation](#) [documentation-tool](#) [openapi](#)  
TypeScript · ⭐ 21.8k · Updated 6 days ago

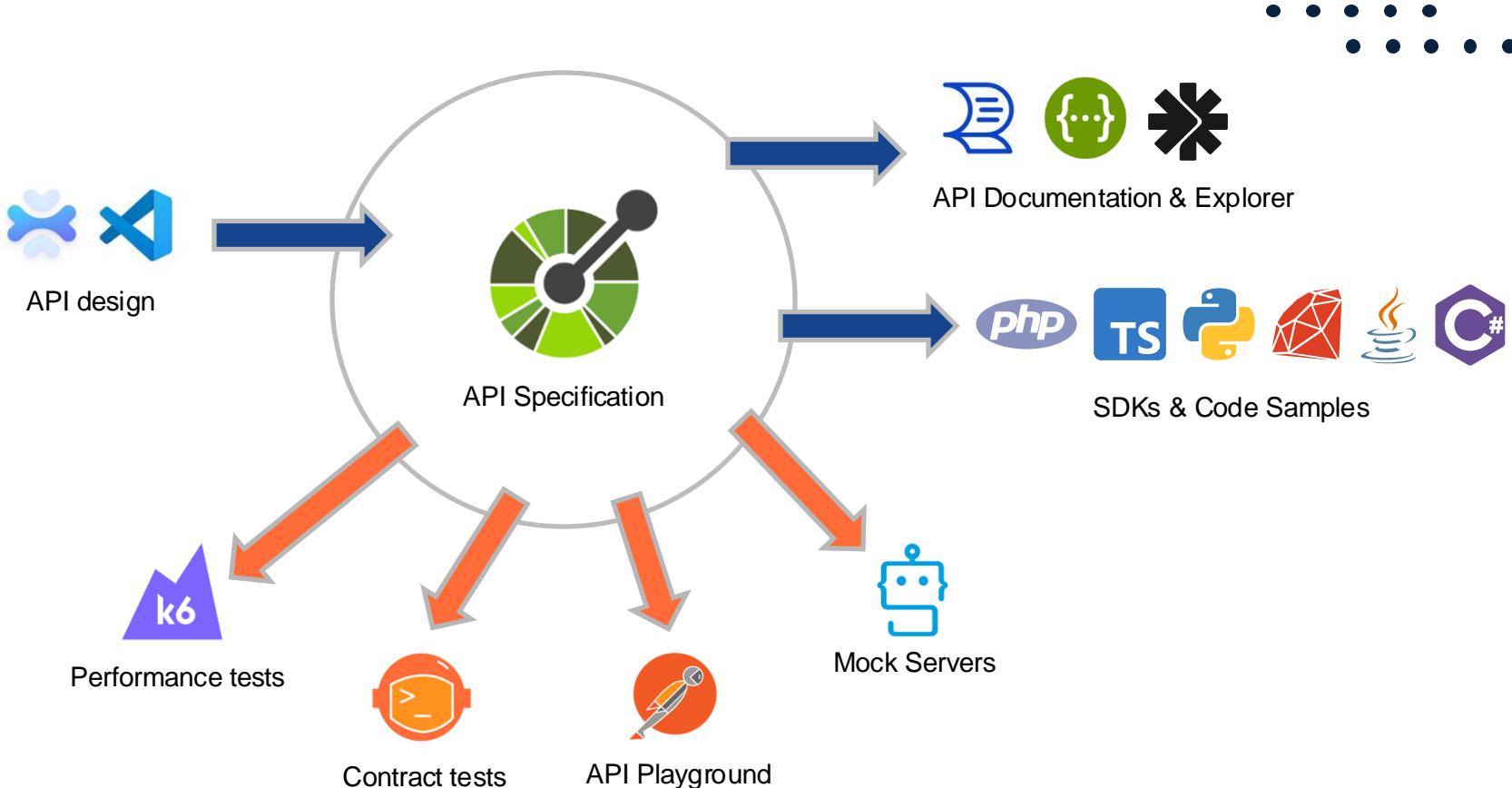
 **APIs-guru/openapi-directory**  
Wikipedia for Web APIs. Directory of REST API definitions in OpenAPI 2.0/3.x format  
[api](#) [aws](#) [azure](#) [rest-api](#) [opendata](#)  
3.5k · Updated 17 hours ago

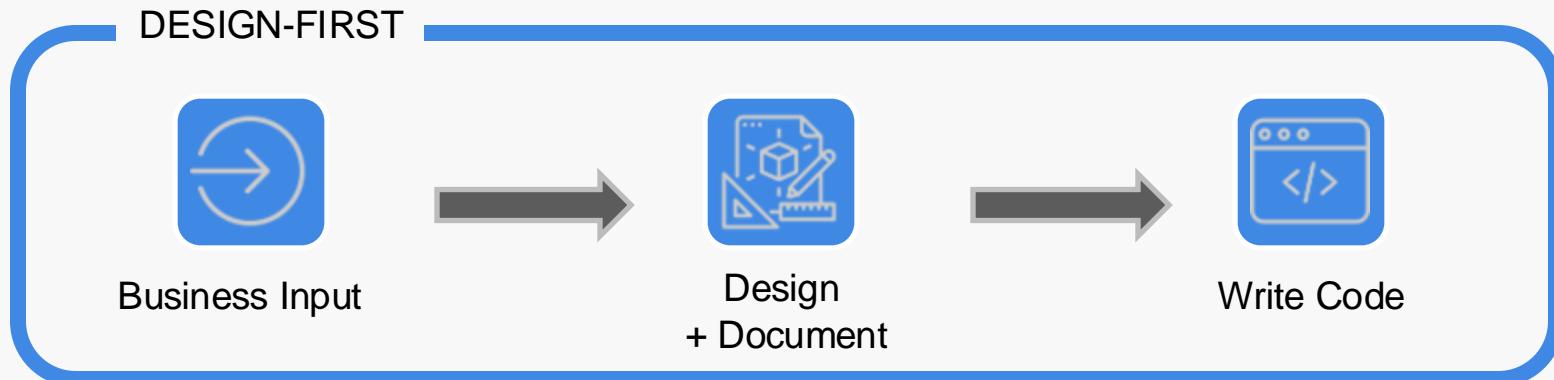
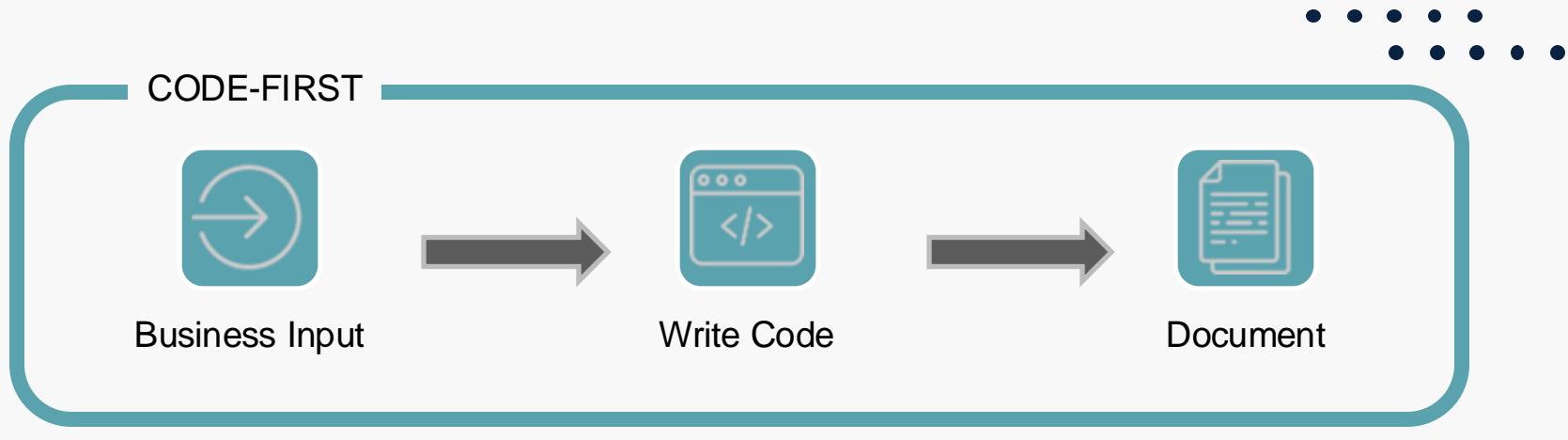
 **stripe/openapi**  
An OpenAPI specification for the Stripe API.  
343 · Updated 3 hours ago

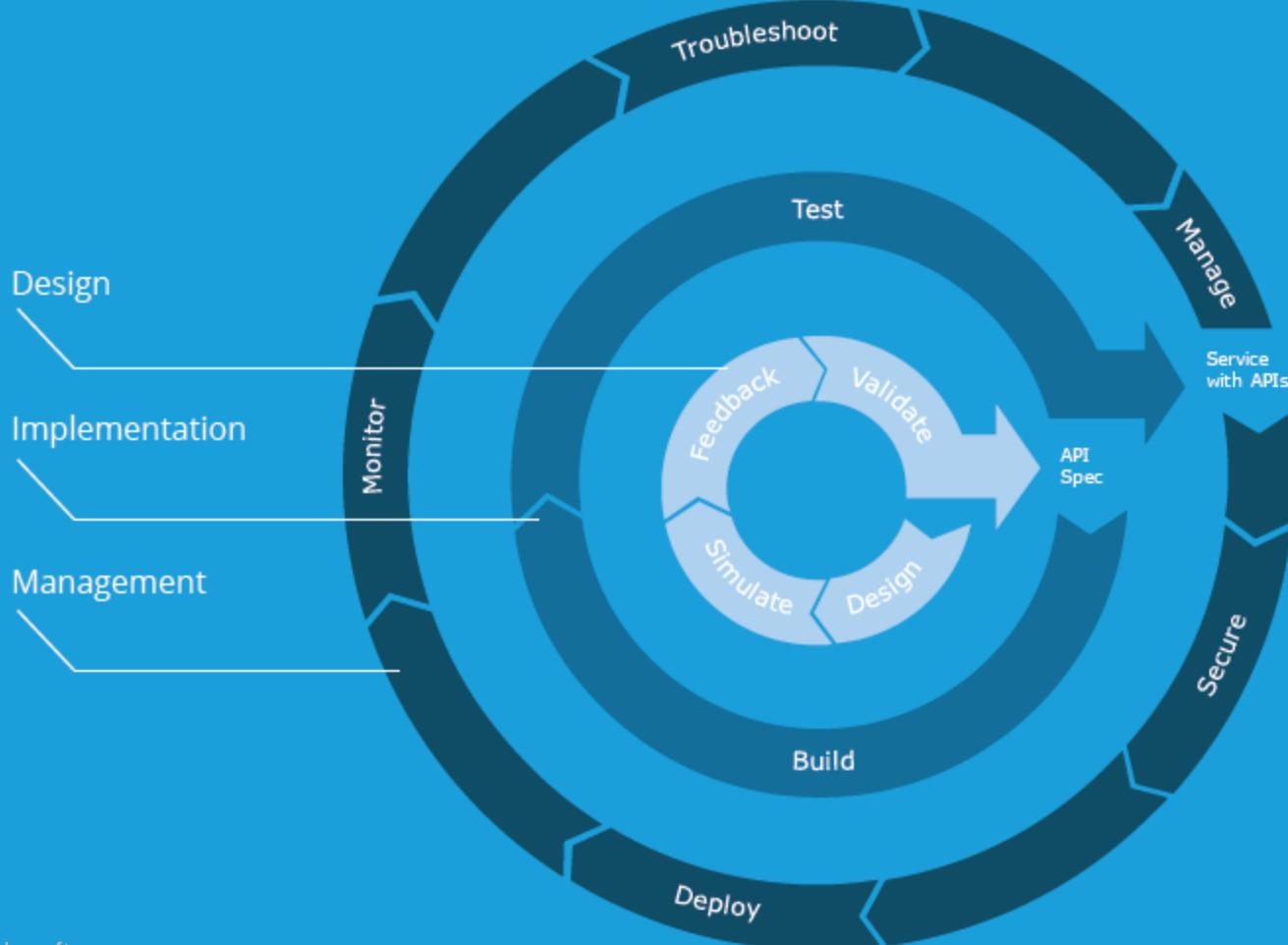
Sponsor open source projects you depend on  
Contributors are working behind the scenes to make open source better for everyone—give them the help and recognition they deserve.  
[Explore sponsorable projects →](#)

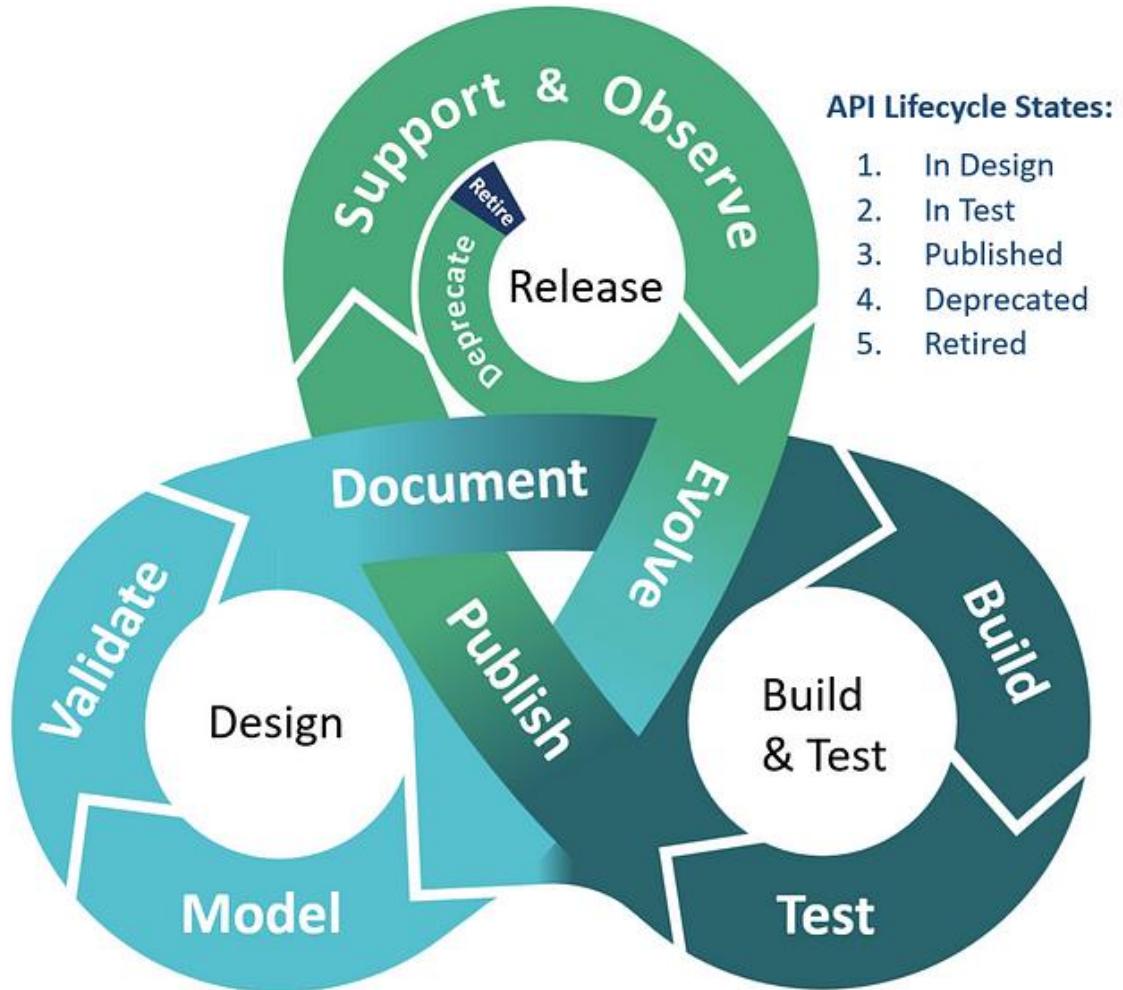
How can we improve search? [Give feedback](#)

ProTip! Press the *I* key to activate the search input again and adjust your query.



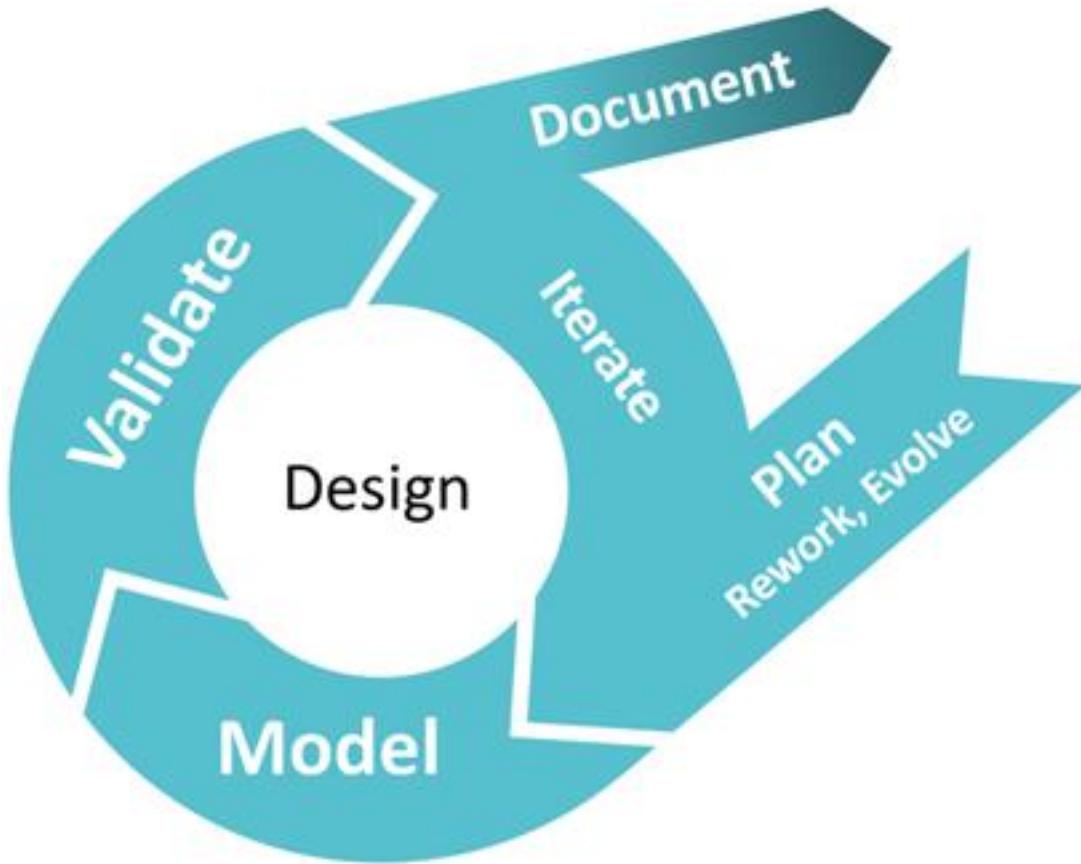






# Spec-driven benefits

- Better **documentation**
  - Clearer **requirements**
  - Shorter **feedback loops**
  - More **consistent API design**
  - Easier Team **alignment**
  - Docs that **match** the API (and vice versa)
  - Much **lower** updating effort
  - Automatable outputs like:
    - **Documentation**
    - **Tests**
    - **Playgrounds**
    - **SDKs**
- • • • •



... : :

# Write OpenAPI



OpenAPI Spec

... • • • •

# JetBrains



The screenshot shows the IntelliJ IDEA IDE interface with the following details:

- Project Structure:** The left sidebar displays the project structure for "marvel-api". The file "openapi.yaml" is currently selected in the editor.
- Editor:** The main editor area shows the content of "openapi.yaml". The code is as follows:

```
1 > openapi: 3.0.3
2   info:
3     title: Marvel Universe API
4     version: 1.0.0
5     description: 'API for retrieving, creating, updating, and deleting information
6       about the Marvel Universe.'
7     x-logo:
8       url: 'https://upload.wikimedia.org/wikipedia/commons/thumb/7/71/Marvel-Comic
9       Book-Logo.png'
10    servers:
11      - url: 'http://localhost:3904/api'
12        description: Production
13    paths:
14      /characters:
15        get:
16          operationId: getCharacters
17          summary: Get all Marvel characters
18          description: Retrieve a list of all Marvel characters.
19          parameters:
20            - $ref: '#/components/parameters/PageParam'
21            - $ref: '#/components/parameters/LimitParam'
22          responses:
23            '200':
24              description: Successful operation
25              content:
26                application/json:
27                  schema:
28                    $ref: '#/components/schemas/CharactersResponse'
29            '400':
30              description: Bad request
31              content:
32                application/json:
33                  schema:
34                    $ref: '#/components/schemas/BadRequestError'
35            '422':
36              description: Unprocessable entity
37              content:
38                application/json:
39                  schema:
40                    $ref: '#/components/schemas/UnprocessableEntityError'
41    tags:
42      - Characters
```

- Right Panel:** The right panel displays the generated API documentation for "Marvel Universe API (1.0.0)". It includes sections for "Characters" and "Get all Marvel characters".
- Bottom Status Bar:** The status bar at the bottom shows the following information: PHP: 8.3 1:1 {}, Language Services, LF, UTF-8, 2 spaces, Schema: OpenAPI 3.0, and main.

# VS Code



<https://marketplace.visualstudio.com/items?itemName=42Crunch.vscode-openapi>

```
swagger: "2.0",
info: {
  description: "Pixi Photo Sharing API",
  version: "1.0.0",
  title: "Pixi App API",
  contact: {
    email: "nicole.becher@owasp.org"
  },
  license: {
    name: "Apache 2.0",
    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
  }
},
tags: [
  {
    name: "admins",
    description: "Secured Admin-only calls"
  },
  {
    name: "users",
    description: "Operations available to regular, logged in users"
  },
  {
    name: "anyone",
    description: "Operations available to anyone"
  }
],
paths: {
  "/api/login": {
    "post": {
      ...
    }
  }
}
```

# Apidog

<https://apidog.com/>

The screenshot shows the Apidog interface for a "Petstore" sample project. The left sidebar includes sections for APIs, Testing, Settings, Share, Schemas, Requests, and Trash. The main area displays the "GET Find pet by ID" endpoint under the "API" tab. The endpoint URL is /pet/{petId}. The "Params" section shows a "Query Params" table with a single entry: Name (param), Type (integer), Example (1), and Description (ID of pet to return). The "Path Params" table shows a single entry: Name (petId), Type (integer), Example (\*), and Description (ID of pet to return). The "Responses" section lists successful operation (200), Invalid ID supplied (400), and Pet not found (404) with a count of 0 public responses. A "Generate from JSON etc." button is available. At the bottom, there's a table for defining response schema fields like id (integer<int64>).

# Stoplight Platform

<https://stoplight.io/studio/>



The screenshot shows the Stoplight Platform studio interface. The top navigation bar includes a user icon, a dropdown for 'studio-demo', a 'Commit' button, and tabs for 'Todos' and 'Operation'. The main area is titled 'Get Todo' and displays the following details:

- URL:** `http://todos.stoplight.io /todos/{todoId}`
- HTTP Methods:** GET, POST, PUT, PATCH, DELETE, HEAD, OPTIONS, TRACE (GET is selected)
- Endpoint description...** (empty field)
- Request Parameters:** + Security, + Header, + Query Param
- Response Body Options:** JSON Request Body, Form Data
- Responses:** + Response, 200 (green dot), 404 (orange dot), 500 (red dot)
- Response 200:** 200 (selected), delete icon
- Response description...** (empty field)
- Headers:** +

The left sidebar shows the project structure:

- Swagger Petstore**
  - Paths
  - Models
  - Embedded Models
- To-dos**
  - Paths**
    - /todos/{todoId}
      - GET (selected)
      - PUT
      - DELETE
    - /todos
      - GET (selected)
      - POST
  - Models
- Common**
  - Models

# Scalar Editor

<https://docs.scalar.com/swagger-editor>



Scalar ▾ Register Login | Star · 5557

Editing Preview

Pages Versions Customize Publish

Guide Reference

Q Search HK

Characters

Get all Marvel characters **GET**

Create a new Marvel character **POST**

Get a Marvel character **GET**

Update a Marvel character **PUT**

Delete a Marvel character **DEL**

Teams

Movies

Models

636        message:  
637            description: Details about the validation error.  
638            type: string  
639        example:  
640            error: Validation errors  
641            details:  
642              - field: first\_name  
643              message: Field 'first\_name' must be of type 'String'.  
644        BadRequestError:  
645            type: object  
646            properties:  
647              error:  
648              description: Description of the error.  
649              type: string  
650        example:  
651              error: Page number cannot be greater than total number of pages  
652        NotFoundError:  
653            type: object  
654            properties:  
655              error:  
656              description: Description of the error.  
657              type: string  
658        example:  
659              error: Marvel character not found  
660        securitySchemes:  
661            bearerAuth:  
662              type: http  
663              scheme: bearer  
664        security:  
665            - bearerAuth: []  
666        tags:  
667            - name: Characters  
668              description: Manage all Marvel characters.  
669            - name: Teams  
670              description: Manage all teams from the Marvel Universe.  
671              x-exclude: true  
672            - name: Movies  
673              description: Manage all movies from the Marvel Universe.  
674              x-exclude: true  
675

Upload File Import URL

1.0.0 OAS 3.0.3

**Marvel Universe API**

[Download OpenAPI Spec](#)

API for retrieving, creating, updating, and deleting information about the Marvel Universe.

BASE URL

Production

BEARER AUTHENTICATION (BEARERAUTH)

Token

CLIENT LIBRARIES

Shell  Ruby  Node.js  PHP  Python  More

CURL Shell

## Characters

Manage all Marvel characters.

ENDPOINTS

**GET** /characters  
**POST** /characters  
**GET** /characters/{id}  
**PUT** /characters/{id}  
**DELETE** /characters/{id}

\* Dark Mode

# API Fiddle

<https://www.api-fiddle.com>



api  
fiddle

Preview OpenAPI

GET /books

Fetches a comprehensive list of all books available in the bookstore. Each book object includes essential details such as the book's ID, name, and publication date.

This endpoint is paginated using the 'page' syntax.

**Parameters**

- title - query  
Filter the list of books according to the title.  
Example: ?title=string  
Type:  
string

- authorId - query  
Filter the list of books according to the authorId  
Example: ?authorId=1  
Type:  
integer

**Responses**

- 200 OK  
A JSON array containing book objects.  
Type:

API Models

GET /books

Fetches a comprehensive list of all books available in the bookstore. Each book object includes essential details such as the book's ID, name, and publication date.

This endpoint is paginated using the 'page' syntax.

**Query parameters**

**title** string or null  
Filter the list of books according to the title.

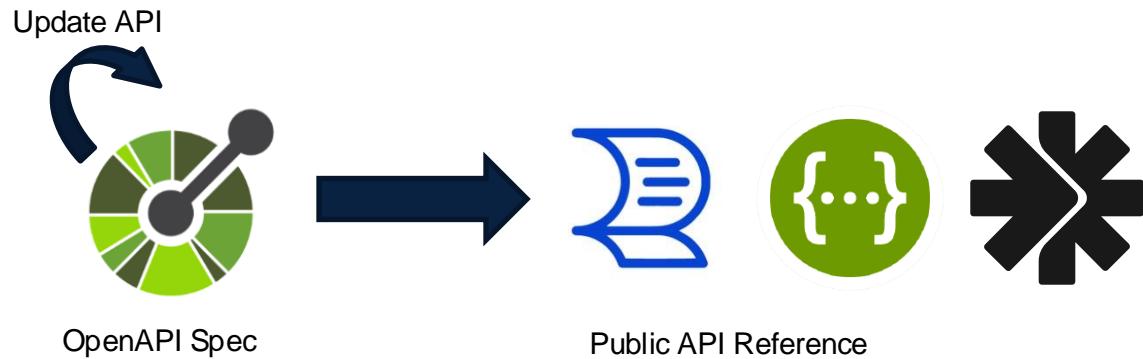
**authorId** integer or null  
Filter the list of books according to the authorId

**Responses**

**200 - Ok** array  
A JSON array containing book objects.

**400 - Bad Request** object  
The server could not understand the request due to invalid syntax. The client should modify the request and try again.

# API Reference



• • • • •

# Redoc(ly)



<https://github.com/Redocly/redoc>

Get all audience configurations

Returns a list of audience configurations. The configurations are returned sorted by name.

**AUTHORIZATIONS:** X-ApiKey

**QUERY PARAMETERS**

- \$top: integer <int32> Default: 100 The number of records to return, used for pagination.
- \$skip: integer <int32> Default: 0 The number of items to skip over before returning items
- \$count: boolean Default: false Request a count of matching items included with the returned results

**HEADER PARAMETERS**

- X-ApiKey: string Default: {{username}}:{{password}} Example: {{username}}:{{password}} API key to authorize requests.

**Responses**

GET /lists/audiences Get all audience configurations

POST Create a audience configuration

RESPONSE SCHEMA: application/json

GET /lists/audiences

**Response samples**

200 422

Content type application/json

```
{  
  "value": [  
    {  
      "id": "12345-6789-01234-567-890",  
      "tenant": "12345-6789-3333-1111-2222",  
      "name": "Master Contact list",  
      "apiName": "master_contact",  
      "systemFields": {  
        "mobile": "MASTER.MOBILE",  
        "language": "MASTER.LANGUAGE"  
      },  
      "type": "Userlist",  
      "requireSegment": true,  
      "created": "2020-01-31T10:31:21.286Z",  
      "brands": [  
        { ... }  
      ]  
    },  
    { ... }  
  ],  
  "@count": 68,  
  "@nextLink": "https://api.selligent.com/v2/lists/audiences?$count=68&$skip=100"  
}
```

# Swagger UI



<https://swagger.io/tools/swagger-ui/>

A Audience lists is used to define the target of a message or journey and personalize the message with data from the list. An audience list contains in general the global contacts details, demographic information, language preferences etc.

## Lists - Audience

The Audience API allows you to create, delete, and update your audience lists. You can retrieve specific audience configurations as well as a list of all your audience configurations.

<code>GET</code>	<code>/lists/audiences</code>	Get all audience configurations	
<code>POST</code>	<code>/lists/audiences</code>	Create a audience configuration	
<code>GET</code>	<code>/lists/audiences/lookup</code>	Lookup a audience configuration	
<code>GET</code>	<code>/lists/audiences/{audienceId}</code>	Get a audience configuration details	
<code>PUT</code>	<code>/lists/audiences/{audienceId}</code>	Update a audience configuration	
<code>DELETE</code>	<code>/lists/audiences/{audienceId}</code>	Destroy a audience configuration	
<code>POST</code>	<code>/lists/audiences/{audienceId}/archive</code>	Archive a audience configuration	

## Lists - Audience - Fields

All endpoints related to the audience fields

<code>GET</code>	<code>/lists/audiences/{audienceId}/fields</code>	Get all fields/schema for an audience.	
<code>PUT</code>	<code>/lists/audiences/{audienceId}/fields</code>	Add or update a collection of fields to an audience	
<code>GET</code>	<code>/lists/audiences/{audienceId}/fields/{fieldId}</code>	Get details for a specific field	
<code>PUT</code>	<code>/lists/audiences/{audienceId}/fields/{fieldId}</code>	Update the field for a specific list	
<code>DELETE</code>	<code>/lists/audiences/{audienceId}/fields/{fieldId}</code>	Delete a specific field	

# Scalar



<https://github.com/scalar/scalar>

## CHARACTERS

Get all Marvel characters GET

Create a new Marvel character POST

Get a Marvel character by ID GET

Update a Marvel character by ID PUT

Delete a Marvel character by ID DELETE

## MODELS

## Create a new Marvel character

Endpoint to create a new Marvel character.

### Body

`id` integer · int64 read-only nullable

ID of the Marvel character

`first_name` string

The first name of the Marvel character.

`last_name` string

The last name of the Marvel character.

`name` REQUIRED string

The full name of the Marvel character.

`description` string

A brief description of the Marvel character.

`powers` array string[]

List of superpowers possessed by the Marvel character.

### Responses

201

Successfully created

422

Bad Request

POST /characters

Shell cURL ↗

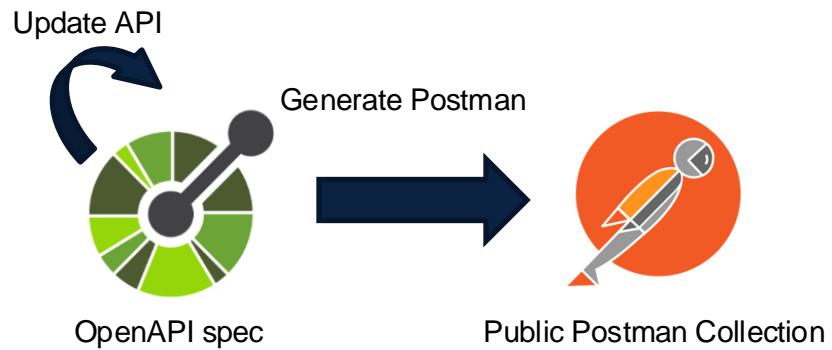
```
1 <?php
2
3 $client = new \GuzzleHttp\Client();
4
5 $response = $client->request('POST', 'http://localhost:3004/
6 'body' => [
7 'first_name' => 'Rocket',
8 'last_name' => 'Raccoon',
9 'name' => 'Rocket Raccoon',
10 'description' => 'A genetically modified raccoon with a knac
11 'powers' => [
12 'Engineering genius',
13 'Expert marksman',
14 'Strategic thinking'
```

TEST REQUEST ↗

201 422

```
{
  "id": 7,
  "first_name": "Tony",
  "last_name": "Stark",
  "name": "Iron Man",
  "description": "Genius, billionaire, playboy, philanthropist",
  "powers": [
    "Superhuman strength",
    "Powered armor suit",
    "Genius-level intellect"
```

# Generate Postman



• • • • •

# Postman



<https://www.postman.com/>

The screenshot shows the Postman application interface. On the left, there's a sidebar with various sections: Collection, Environment, APIs, Mock Servers, Monitors, Flows, and History. The main area displays a collection titled "Notion API / Databases / Retrieve a database". This collection contains several requests:

- A GET request to <https://api.notion.com/v1/databases/:id>. It includes a "Params" tab with a "Query params" table:

KEY	VALUE	DESCRIPTION
Key	Value	Description

and a "Path variables" table:

KEY	VALUE	DESCRIPTION
id	<code>((DATABASE_ID))</code>	Required. Enter database id.
- A 200 Success- Retire... response.
- A POST request to Query a database.
- A POST request to Create a database.
- A PATCH request to Update a database.
- Other requests include Pages, Users, Blocks, and Comments.

On the right side, there's a "Documentation" panel for the "Retrieve a database" request, which provides the URL <https://api.notion.com/v1/databases/:id>, a description of the endpoint, and information about authorization. There's also a "Request Header" section showing the header `Notion-Version 22-02-22` and a "Path Variables" section mapping the variable `id` to the value `((DATABASE_ID))`.

At the bottom, there are navigation links like "Find and Replace", "Console", "Cookies", "Capture Requests", "Runner", "Trash", and "Help".

# Bruno



<https://www.usebruno.com>

The screenshot shows the Bruno application window. At the top, there's a navigation bar with a logo, user name, and tabs for 'Collections' and 'sample-collection'. Below the navigation is a search bar. The main area displays an API request configuration for a 'Users' endpoint. The request method is 'GET', the URL is 'https://reqres.in/api/users?page=2', and the 'Params' tab is selected, showing a single parameter 'page' with a value of '2'. To the right, the 'Response' tab shows the JSON response from the API call. The response includes pagination details and two user objects. The entire interface has a modern, dark-themed design with a pink-to-purple gradient background.

```
1 ▼ {
2   "page": 2,
3   "per_page": 6,
4   "total": 12,
5   "total_pages": 2,
6   "data": [
7     {
8       "id": 7,
9       "email": "michael.lawson@reqres.in",
10      "first_name": "Michael",
11      "last_name": "Lawson",
12      "avatar": "https://reqres.in/img/faces/7-
image.jpg"
13    },
14    {
15      "id": 8,
16      "email": "lindsay.ferguson@reqres.in",
17      "first_name": "Lindsay",
18      "last_name": "Ferguson",
19      "avatar": "https://reqres.in/img/faces/8-
```

# .HTTP Files



JetBrains / VS Code

The screenshot shows the 'marvel-api' project in the 'publish-postman' directory. The current file is 'marvel.postman.http'. The code editor displays a Postman collection for the Marvel Universe API. It includes two main requests: 'GET {{baseUrl}}/characters?' and 'POST {{baseUrl}}/characters'. The 'GET' request retrieves a list of characters with parameters 'page=1&limit=10', an Authorization header with 'Bearer {{bearerToken}}', and an Accept header of 'application/json'. It includes examples for successful and bad request responses. The 'POST' request creates a new character with the same headers and a JSON body containing 'name: "Rocket Raccoon"'. It also includes examples for successful creation and bad request responses. The interface features a dark theme with various toolbars and panels on the right side.

```
# Marvel Universe API

##
# group: Characters
# @name Get all Marvel characters
# Retrieve a list of all Marvel characters.
GET {{baseUrl}}/characters?
  page=1&
  limit=10
Authorization: Bearer {{bearerToken}}
Accept: application/json

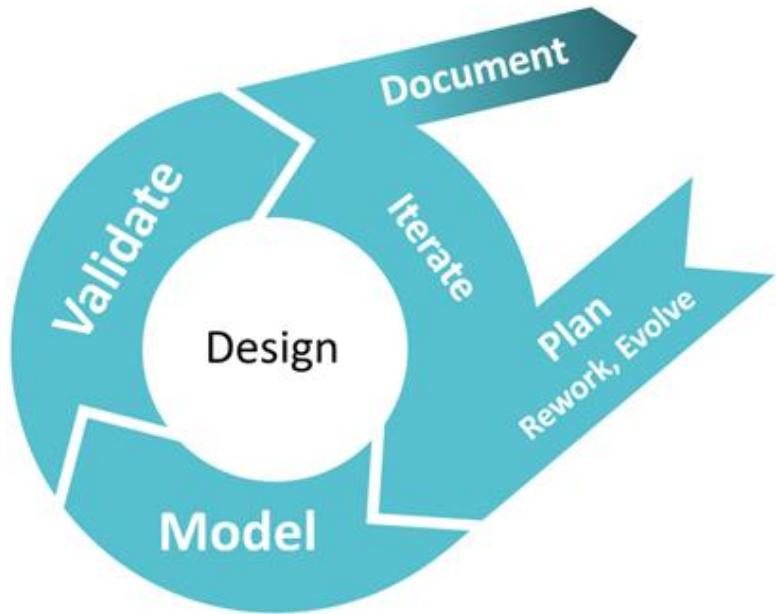
  <> ./marvel.postman_responses/successful_operation.200.json
// {{baseUrl}}/characters?page=1&limit=10
  <> ./marvel.postman_responses/bad_request.400.json
// {{baseUrl}}/characters?page=1&limit=10

##
# group: Characters
# @name Create a new Marvel character
# Endpoint to create a new Marvel character.
POST {{baseUrl}}/characters
Authorization: Bearer {{bearerToken}}
Content-Type: application/json
Accept: application/json

  {"name": "Rocket Raccoon"...

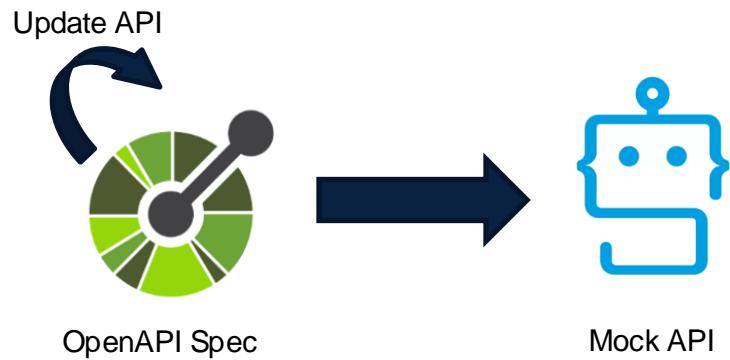
  <> ./marvel.postman_responses/successfully_created.201.json
// {{baseUrl}}/characters
  <> ./marvel.postman_responses/bad_request.422.json
// {{baseUrl}}/characters

##
# group: Characters
# @name Get a Marvel character
```



• • • • •

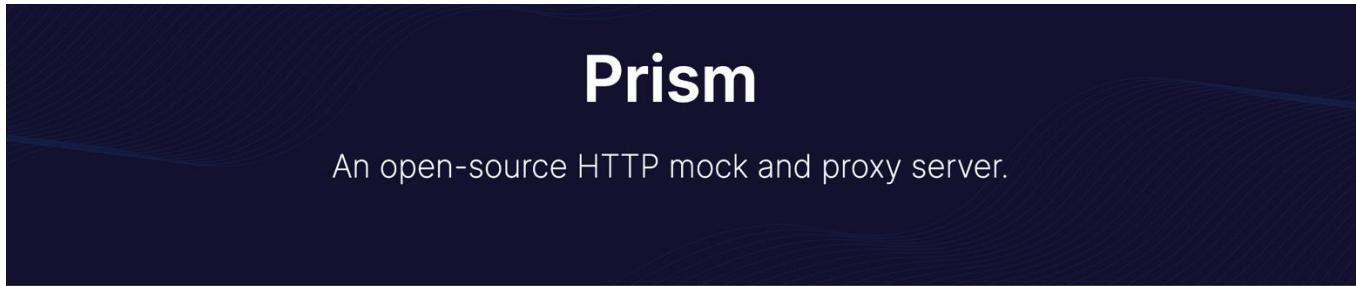
# Mock API



# Prism



<https://github.com/stolightio/prism>



The image shows the top portion of the Prism landing page. It features a dark blue background with light blue wavy patterns. In the center, the word "Prism" is written in a large, white, sans-serif font. Below it, the text "An open-source HTTP mock and proxy server." is displayed in a smaller, white, sans-serif font.

Accelerate API development with realistic mock servers, powered by OpenAPI documents.

[Start Mocking for Free](#)

[View on GitHub](#)

[View Docs](#)



Quick Iterations



Develop in Parallel



Dynamic Examples



Validation



Mocking Callbacks

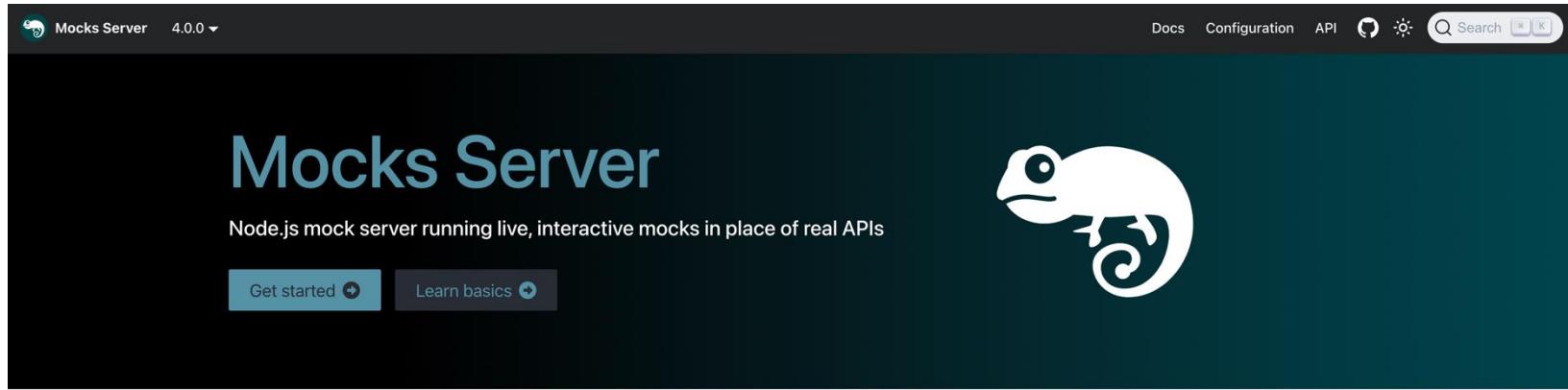


Proxy

# Mocks Server



<https://www.mocks-server.org/>



The screenshot shows the official website for Mocks Server. At the top, there's a navigation bar with the logo, version 4.0.0, Docs, Configuration, API, and a search bar. The main title "Mocks Server" is prominently displayed in large blue text. Below it, a subtitle reads "Node.js mock server running live, interactive mocks in place of real APIs". There are two buttons: "Get started" and "Learn basics". To the right is a large white chameleon logo. The background is dark teal.



## Quick and simple

Install it using NPM or Docker. Add files defining responses for the API routes. Start it with a single command. Change responses using settings, even while it is running.

[Learn more →](#)



## Flexible

Define routes using `YAML`, `JSON`, `JavaScript` or `TypeScript`. Configure Babel at your convenience for reading files. Or define routes programmatically. HTTP and HTTPS protocols are supported.

[Learn more →](#)



## Extensible

Use Express middlewares in routes. Define custom handlers for the routes. Add custom Express routers. Or create a plugin and have full access to the core JavaScript API.

[Learn more →](#)



## Easy to control

Control it using the interactive CLI, or use the administration REST API, or start it using JavaScript and control everything. Other integrations are available, such as Cypress commands.

[Learn more →](#)

# Microcks



<https://microcks.io/>

SERVICES & APIs > Marvel API - 1.0

## Marvel API - 1.0

Created 2 minutes ago, last update on Aug 30, 2024, 4:25:34 PM

Edit Labels

**PROPERTIES**

Service Type: DIRECT REST

Contracts: Swagger 2.0 spec ↗  
OpenAPI 3.0 spec ↗

Statistics: Mocks invocations ↗

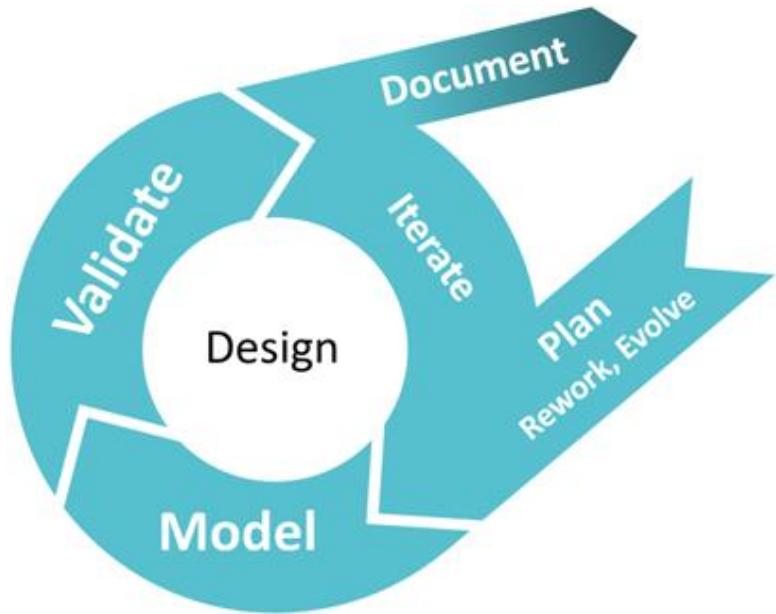
**Tests**

Conformance index: A B C D E → 0.0% Conformance score

You may have already used this Direct API to record resources.  
Check what's in the repository ↗.

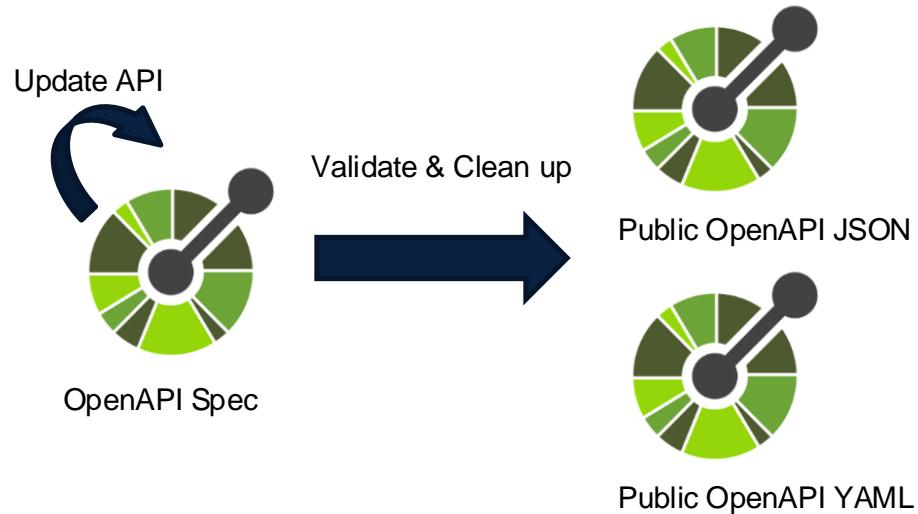
**OPERATIONS**

Method	Path	Status	Sample(s)
GET	/Movies	GET	0 sample(s)
POST	/Movies	POST	0 sample(s)
DELETE	/Movies/:id	DELETE with URI_PARTS dispatcher	0 sample(s)
GET	/Movies/:id	GET with URI_PARTS dispatcher	0 sample(s)
PUT	/Movies/:id	PUT with URI_PARTS dispatcher	0 sample(s)



• • • • •

# Validate & Publish OpenAPI



# Spectral



<https://stoplight.io/open-source/spectral>

# Spectral

An open-source API style guide enforcer and linter.

Why write API Style Guides as lengthy manifestos when you could automate them? Make sure APIs are secure, consistent, useful. Spectral is open-source, but is also baked into Stoplight, with extensions for VS Code and other integration options, giving you real-time feedback wherever you design APIs.

[Start Linting for Free](#)

[View on GitHub](#)

[View Docs](#)



Style Guides



Improve API Quality



CLI & JavaScript



Stoplight Platform



Style Guide Projects

# Redocly CLI



<https://redocly.com/redocly-cli/>

The screenshot shows the Redocly CLI dashboard. At the top, there's a navigation bar with the Redocly logo, 'Products', 'Docs', 'Pricing', 'Enterprise', 'About', 'Login', and 'Sign Up' buttons. Below the navigation bar, the main content area has a title 'main#1' and a timestamp 'Thu Aug 19 2021'. It shows a status badge with a green checkmark and the word 'published'. A button labeled 'GET SNAPSHOT' is visible. Below this, a progress bar indicates four steps: 'Provision', 'Validate and Build', 'Assets upload', and 'Add to registry', each marked with a green checkmark. A green success message at the bottom says 'Woohoo! Your OpenAPI definition is valid.' A 'RESTART' button is also present. At the bottom of the main section, a 'Results' summary states 'No errors or warnings found.' and includes a 'SHOW TERMINAL' button.

## Validate a multi-file definition without having to bundle it

Behemoth YAML gets old pretty quick. An unbundled definition is far easier to edit and check for duplicates. Our Redocly CLI lets you quickly validate and lint a multi-file definition before bundling it into one file for distribution.

# Vacuum



<https://github.com/daveshanley/vacuum>



**vacuum - The world's fastest OpenAPI & Swagger linter.**



An **ultra-super-fast**, lightweight OpenAPI linter and quality checking tool, written in golang and inspired by [Spectral](#).

# OpenAPI-Format



<https://www.npmjs.com/package/openapi-format>

npm Search packages Search Sign Up

---

**openapi-format**  
1.16.0 • Public • Published 15 days ago

[Readme](#) [Code](#) Beta [4 Dependencies](#) [4 Dependents](#) [48 Versions](#)

---

 **OPENAPI-FORMAT**

npm v1.16.0 downloads 21k/week

---

## openapi-format

Format an OpenAPI document by ordering, formatting and filtering fields.

The `openapi-format` CLI can load an OpenAPI file, sorts the OpenAPI fields by ordering them in a hierarchical order, format the casing of the fields and can output the file with clean indenting, to either JSON or YAML.

Next to the ordering & formatting, the CLI provides additional options to filter fields & parts of the OpenAPI document based on flags, tags, methods, operationID's and even unused components.

---

Install  
`> npm i openapi-format`

---

Repository  
[github.com/thim81/openapi-format](#)

---

Homepage  
[github.com/thim81/openapi-format#re...](#)

---

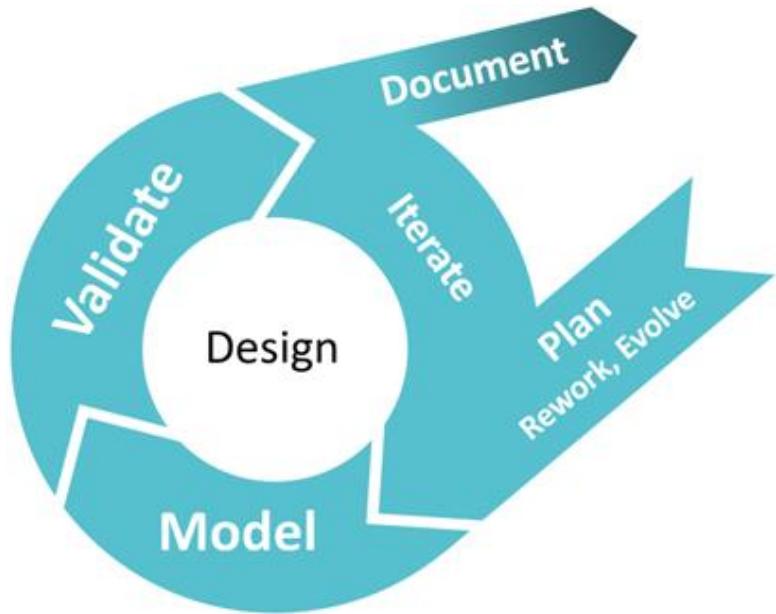
Weekly Downloads  
20,761 

---

Version	License
1.16.0	MIT

---

Unpacked Size	Total Files
127 kB	16



• • • • •

# Publish Docs

Update API



OpenAPI Spec

Publish



Public API References



Public Postman Collection



Public OpenAPI JSON/YAML

• • • • •





<https://redocly.com>

The screenshot shows the Redocly homepage with a dark blue background. At the top left is the Redocly logo (a stylized 'R' icon) and the word "Redocly". To its right are navigation links: "Products", "Docs", "Pricing", "Enterprise", and "About". On the far right are "Login" and "Sign Up" buttons. A large white text area in the center-left contains the headline "Make API docs your superpower". Below it is a subtext: "Beautiful API documentation loved by teams and API consumers. Brought to you by the open-source extraordinaires behind Redoc." At the bottom left is an orange "Sign Up" button. In the bottom right corner, there's a graphic of three overlapping purple semi-transparent cards showing icons related to API development: a gear, a code editor, and a document. At the very bottom, logos for various companies are listed: GitHub, Adobe, Hewlett Packard Enterprise, Magento, T-Mobile, TripAdvisor, Checkr, and Trend Micro.

Redocly

Products Docs Pricing Enterprise About

Login Sign Up

# Make API docs your superpower

Beautiful API documentation loved by teams and API consumers.  
Brought to you by the open-source extraordinaires behind Redoc.

Sign Up

Redocly powers 10,000+ companies that power the API economy

GitHub Adobe Hewlett Packard Enterprise Magento  
 T-Mobile TripAdvisor Checkr Trend Micro

# Readme.com



<https://readme.com/>

The screenshot shows the Hoot API documentation page on the Readme.com platform. The top navigation bar includes links for 'Hoot Blog' and 'We're Hiring!', along with dropdown menus for 'v2.0' and 'Product Guide', and buttons for 'API Reference', 'Changelog', and 'Community Discussions'. A search bar is also present.

The main content area features a section titled 'Send an API Log to Hoot' with a 'POST' button and a URL 'https://metrics.hoot.io'. Below this, there's a 'YOUR REQUEST HISTORY' section showing '0 Calls' over '7 Days'. The 'BODY PARAMS' section has a 'ADD OBJECT' button. The 'RESPONSE' section displays a green '202' status with the message 'Finished saving (not necessarily successful)'. A note indicates the response was 'Updated over 1 year ago'.

On the right side, there are sections for 'LANGUAGE' (Shell, Node, Ruby, PHP, Python), 'AUTHENTICATION' (Basic, username:password, BASIC, BASE64), and a 'CURL' request example:

```
curl --request POST \
--url https://metrics.hoot.io/request \
--header 'Content-Type: application/json'
```

A 'Try It!' button is located next to the CURL example. Below it, a 'RESPONSE' section is shown with the instruction 'Click Try It! to start a request and see the response here!'. At the bottom of the page, there's a feedback poll: 'Did this page help you?' with 'Yes' and 'No' options, and a 'Powered by readme' footer.



<https://squidfunk.github.io/mkdocs-material>

The screenshot shows the GitHub repository page for 'squidfunk/mkdocs-material'. The header includes the repository name, a star icon (9.5.3), a fork icon (16.8k), and a file icon (3.2k). Below the header is a navigation bar with links to Home, Getting started, Setup, Plugins, Reference, Insiders, and Blog. The main content area displays the 'Getting started' page.

Getting started  
Installation  
Creating your site  
Publishing your site  
Customization  
Conventions  
Browser support  
Enterprise feedback ⓘ  
Philosophy  
Alternatives  
License  
Changelog  
How to upgrade  
  
Contributing  
Reporting a bug  
Reporting a docs issue  
Requesting a change  
Adding translations  
Making a pull request  
Asking a question

## Getting started



Table of contents  
Installation  
with pip  
with docker  
with git

Material for MkDocs is a powerful documentation framework on top of [MkDocs](#), a static site generator for project documentation.<sup>1</sup> If you're familiar with Python, you can install Material for MkDocs with [pip](#), the Python package manager. If not, we recommend using [docker](#).

### Installation

#### with pip recommended

Material for MkDocs is published as a [Python package](#) and can be installed with [pip](#), ideally by using a [virtual environment](#). Open up a terminal and install Material for MkDocs with:

Latest 9.x

`pip install mkdocs-material`



This will automatically install compatible versions of all dependencies: [MkDocs](#), [Markdown](#), [Pygments](#) and [Python Markdown Extensions](#). Material for MkDocs always strives to support the latest versions, so there's no need to install those packages separately.

# Stoplight



<https://stoplight.io>

Stoplight

← Home

Example Project

Connect to Edit

master

APIS

Test

- Get User Info by User ID GET
- Update User Information PATCH
- Create New User POST
- Schemas >

Swagger2

- Get User Info by User ID GET
- Update User Information PATCH
- Create New User POST
- Schemas >

SCHEMAS

ProjectModel ●

Invite to Workspace

user@stoplight.io >

powered by Stoplight

Search Example Project...

Changelog

## Get User Info by User ID

Retrieve the information of the user with the matching user ID.

### Request

#### Path Parameters

`userId` integer required  
Id of an existing user.

#### Responses

200 404

User Found

#### Body

application/json

```
1 {
2   "id": 142,
3   "firstName": "Alice",
4   "lastName": "Smith",
5   "email": "alice.smith@gmail.com",
6   "dateOfBirth": "1997-10-31",
7   "emailVerified": true,
8   "signUpDate": "2019-08-24"
9 }
```

# Postman



## Query Params

<b>page</b>	1	The page number for pagination.
<b>limit</b>	10	The number of items to return per page.

## Example

## Successful operation ✓

## Request

```
cURL  
  
curl --location 'http://localhost:3004/characters?page=1&limit=10' \  
--header 'Authorization: Bearer <token>' \  
--header 'Accept: application/json'
```

## Response

## Body Headers (1)

200 OH

```
json
{
  "characters": [
    {
      "name": "Iron Man",
      "id": 7,
      "first_name": "Tony",
      "last_name": "Stark",
      "description": "Genius, billionaire, playboy, philanthropist.",
      "powers": [
        "Superhuman strength"
      ]
    }
  ]
}
```

JUMP TO

## Introduction

- v Characters
    - GET** Get all Marvel characters
    - POST** Create a new Marvel character
    - GET** Get a Marvel character
    - PUT** Update a Marvel character
    - DEL** Delete a Marvel character

# Microcks



<https://microcks.io/>

SERVICES & APIs > Marvel API - 1.0

## Marvel API - 1.0

Created 2 minutes ago, last update on Aug 30, 2024, 4:25:34 PM

Edit Labels

**PROPERTIES**

Service Type: DIRECT REST

Contracts: Swagger 2.0 spec ⓘ, OpenAPI 3.0 spec ⓘ

Statistics: Mocks invocations ⓘ

**Tests**

Conformance index: A B C D E

Conformance score: 0.0 % ➔

Learn More ⓘ

You may have already used this Direct API to record resources.  
Check what's in the repository ⓘ.

**OPERATIONS**

Method	Path	Status	Sample(s)
GET	/Movies	GET	0 sample(s)
POST	/Movies	POST	0 sample(s)
DELETE	/Movies/:id	DELETE with URI_PARTS dispatcher	0 sample(s)
GET	/Movies/:id	GET with URI_PARTS dispatcher	0 sample(s)
PUT	/Movies/:id	PUT with URI_PARTS dispatcher	0 sample(s)

# Backstage



<https://backstage.io/>

Backstage

## APIs

Backstage API Explorer

Search

Home

APIs

Docs

Explore

Tech Radar

Cost Insights

GraphQL

Settings

Type: all

PERSONAL

- Owned: 0
- Starred: 0

BACKSTAGE

- All: 9

OWNER

LIFECYCLE

TAGS

### All apis (9)

NAME	SYSTEM	OWNER	TYPE	LIFECYCLE	DESCRIPTION	TAGS	ACTIONS	
hello-world		team-c	grpc	deprecated	Hello World...			
hello-world-tcp		team-c	tcp	experimental	Hello World...			
petstore		team-c	openapi	experimental	The petstore API	store, rest		
petstore-webhook		team-c	openapi	experimental	The petstore...	store, rest		
spotify		team-a	openapi	production	The Spotify web...	spotify, rest		
starwars-graphql		team-b	graphql	production	SWAPI GraphQL...			
streetlights		team-c	asyncapi	production	The...	mqtt		
wayback-archive		team-a	openapi	production	Archive APIs for...			
wayback-search		team-a	openapi	production	Search API for...			

# EventCatalog



<https://www.eventcatalog.dev/>

EventCatalog

Search EventCatalog  ⌘K

Docs Visualiser Discover

**Domains**

**Orders** v0.0.2

- Orders Domain
- Node Graph
- Order

**Payment** v0.0.2

- Overview

**Services**

**Inventory Service** v0.0.2

- EDA Architecture overview

**Notifications** v0.0.2

- Why this service was created?
- Who owns this service?

**Orders Service** v0.0.2

- Why this service was created?
- Who owns this service?

**Payment Service** v0.0.1

- Why this service was created?
- Who owns this service?

**Messages**

**Inventory adjusted** v0.0.4

- Overview
- Event Details
- Event Name
- Description

## Orders (v0.0.2)

[Go Domain](#) [New domain](#)

### Orders Domain

The Orders domain handles all operations related to customer orders, from creation to fulfillment. This documentation provides an overview of the events and services involved in the Orders domain, helping developers and stakeholders understand the event-driven architecture.

ⓘ Please ensure all services are updated to the latest version for compatibility and performance improvements.

### Node Graph

Download visual

[View in Visualiser](#)

Services (3)

**InventoryService** v0.0.2 Service

**NotificationService** v0.0.2 Service

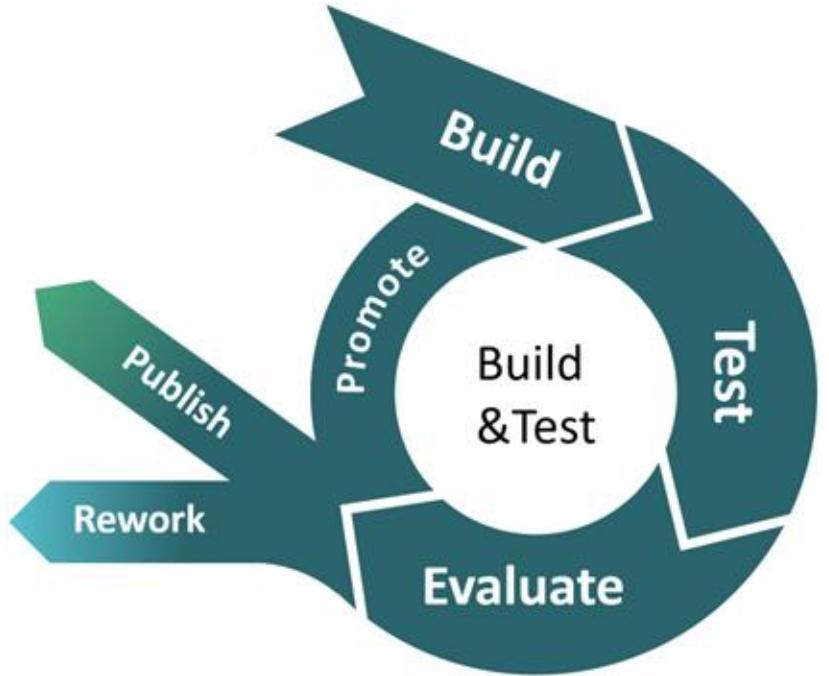
**OrdersService** v0.0.2 Service

Service Owners (1)

**David Boyne** Lead developer

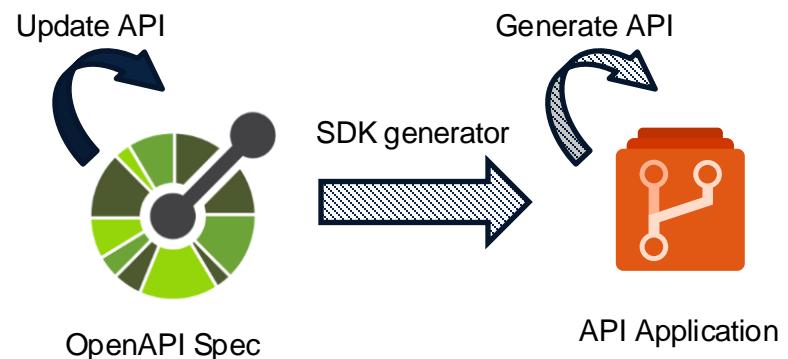
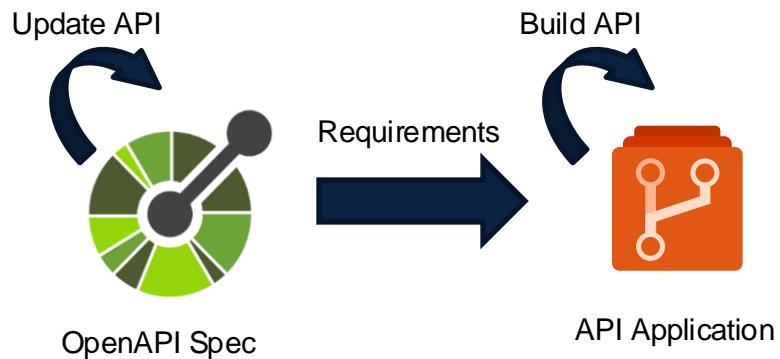
Versions (2)

v0.0.2 (latest) v0.0.1



• • • • •

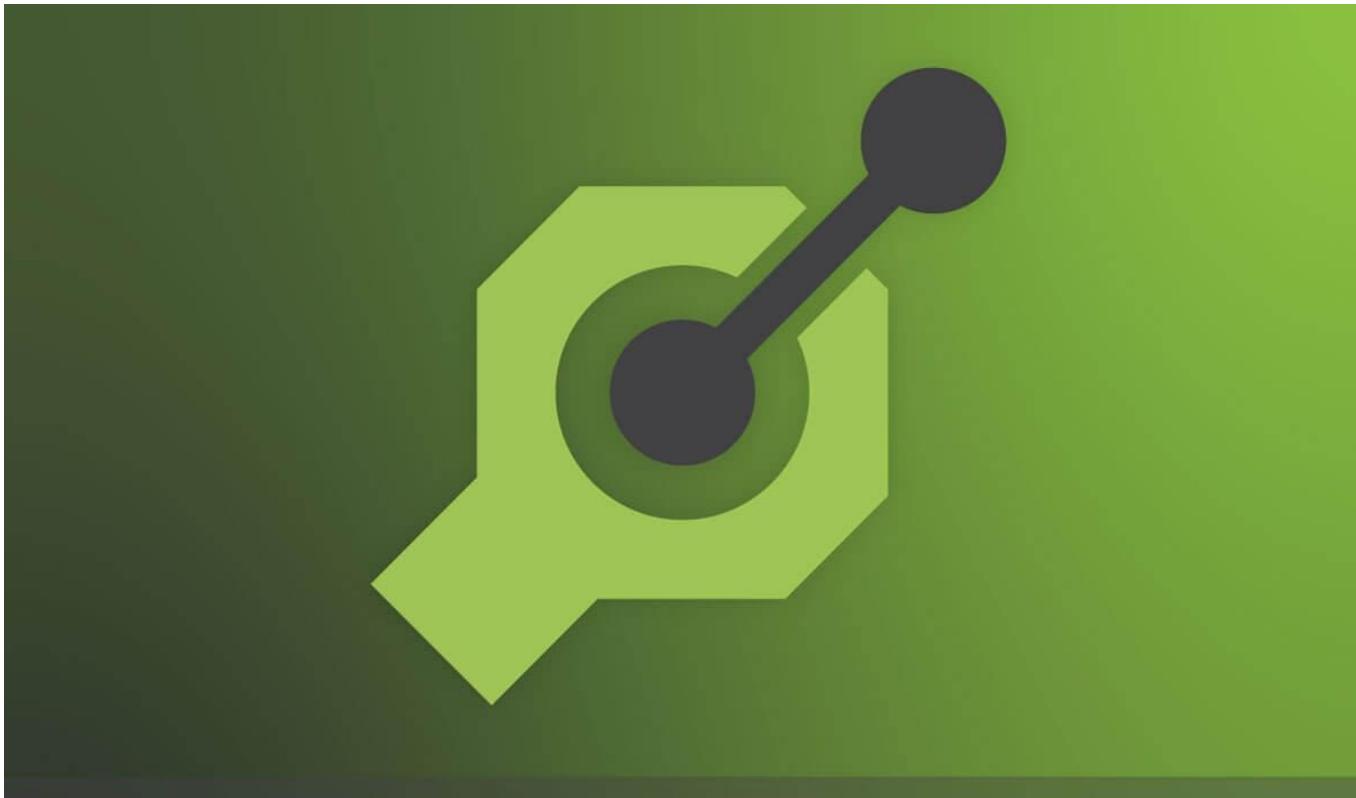
# Build

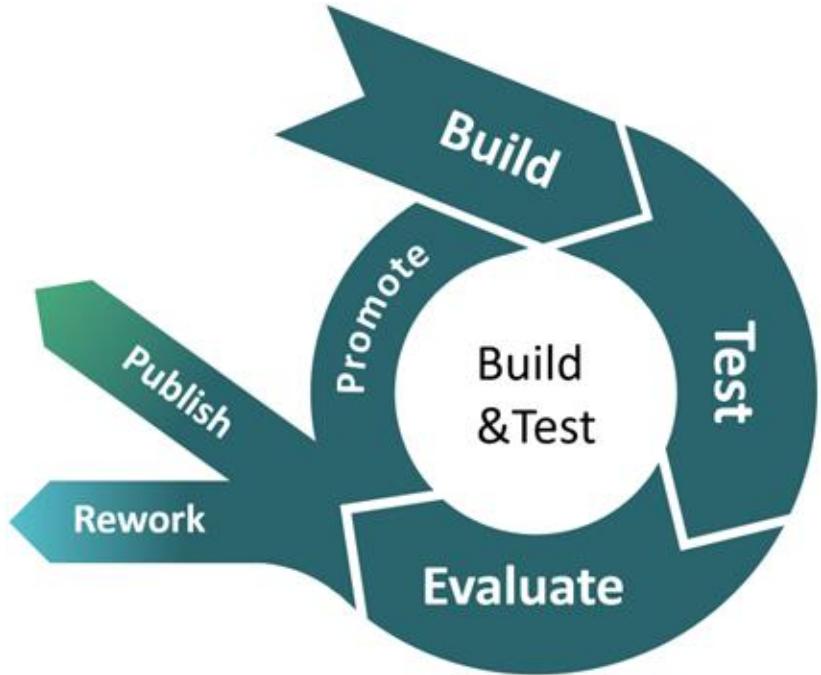


# Server Generation



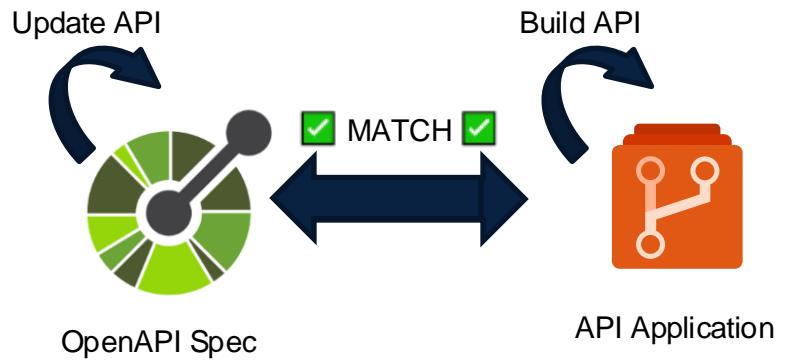
<https://openapi-generator.tech>





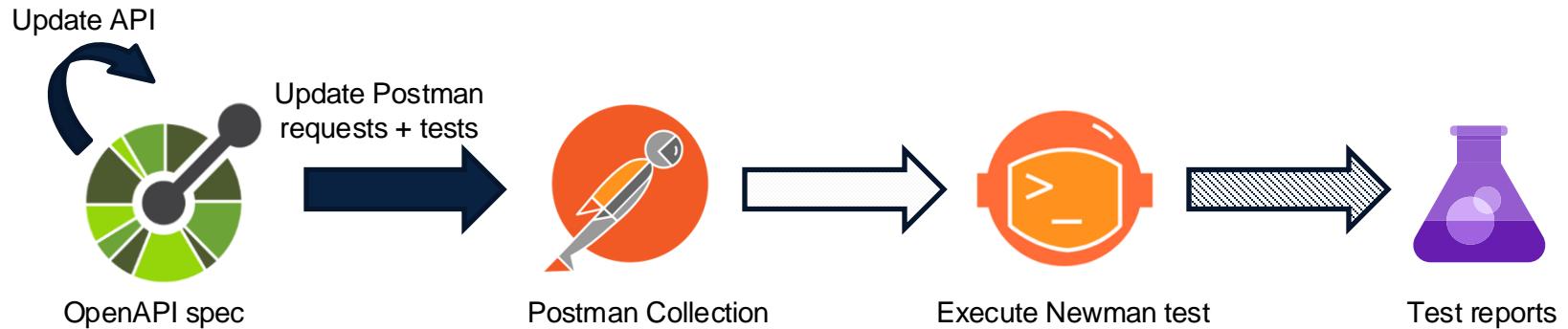
• • • • •

# Contract testing



• • • • •

# Contract testing



• • • • •

# Portman



<https://www.npmjs.com/package/@apideck/portman>

**npm** Search packages

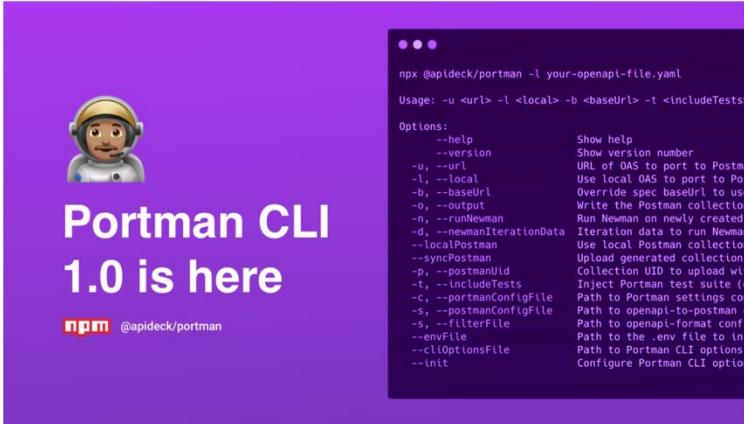
Sign Up Sign In

---

**@apideck/portman** ts

1.26.2 • Public • Published 3 days ago

[Readme](#) [Code](#) Beta [24 Dependencies](#) [2 Dependents](#) [80 Versions](#)



A purple banner featuring a cartoon character wearing a flight helmet and goggles, with the text "Portman CLI 1.0 is here". Below it is the npm logo and "@apideck/portman". To the right is a terminal window showing the Portman CLI usage and options.

```
npx @apideck/portman -l your-openapi-file.yaml
Usage: -u <url> -l <local> -b <baseUrl> -t <includeTests>
Options:
  --help           Show help
  --version        Show version number
  -u, --url        URL of OAS to port to Postman
  -l, --local      Use local OAS to port to Postman
  -b, --baseUrl   Override spec baseUrl to use
  -o, --output     Write the Postman collection
  -n, --runNewman  Run Newman on newly created
  --newmanIterationData  Iteration data to run Newman
  --localPostman   Use local Postman collection
  --syncPostman    Upload generated collection
  -p, --postmanUid Collection UID to upload with
  -t, --includeTests Inject Postman test suite (disabled)
  -c, --portmanConfigFile Path to Portman settings config file
  -s, --postmanConfigFile Path to openapi-to-postman config file
  -s, --filterFile  Path to openapi-format config file
  -e, --envFile    Path to the .env file to inject environment variables
  --cliOptionsfile Path to Portman CLI options file
  --init           Configure Portman CLI options
```

[npm v1.26.2](#) [downloads 9.1k/week](#)

**Portman** 

---

Install [npm i @apideck/portman](#)

---

Repository [github.com/apideck-libraries/portman](#)

---

Homepage [apideck.com](#)

---

Weekly Downloads  9,097

---

Version	License
1.26.2	Apache-2.0

---

Unpacked Size	Total Files
848 kB	301

---

# Newman



<https://www.npmjs.com/package/newman>

**npm**  **Search** [Sign Up](#)

---

**newman** 

6.1.0 • [Public](#) • Published a month ago

[Readme](#) [Code](#) Beta [21 Dependencies](#) [182 Dependents](#) [166 Versions](#)

---

 **POSTMAN**

*Manage all of your organization's APIs in Postman, with the industry's most complete API development environment.*

**newman** *the cli companion for postman*  

Newman is a command-line collection runner for Postman. It allows you to effortlessly run and test a Postman collection directly from the command-line. It is built with extensibility in mind so that you can easily integrate it with your continuous integration servers and build systems.

---

**Install**

```
> npm i newman
```

---

**Repository**

 [github.com/postmanlabs/newman](https://github.com/postmanlabs/newman)

---

**Homepage**

 [github.com/postmanlabs/newman](https://github.com/postmanlabs/newman)

---

**Weekly Downloads**

580,509 

---

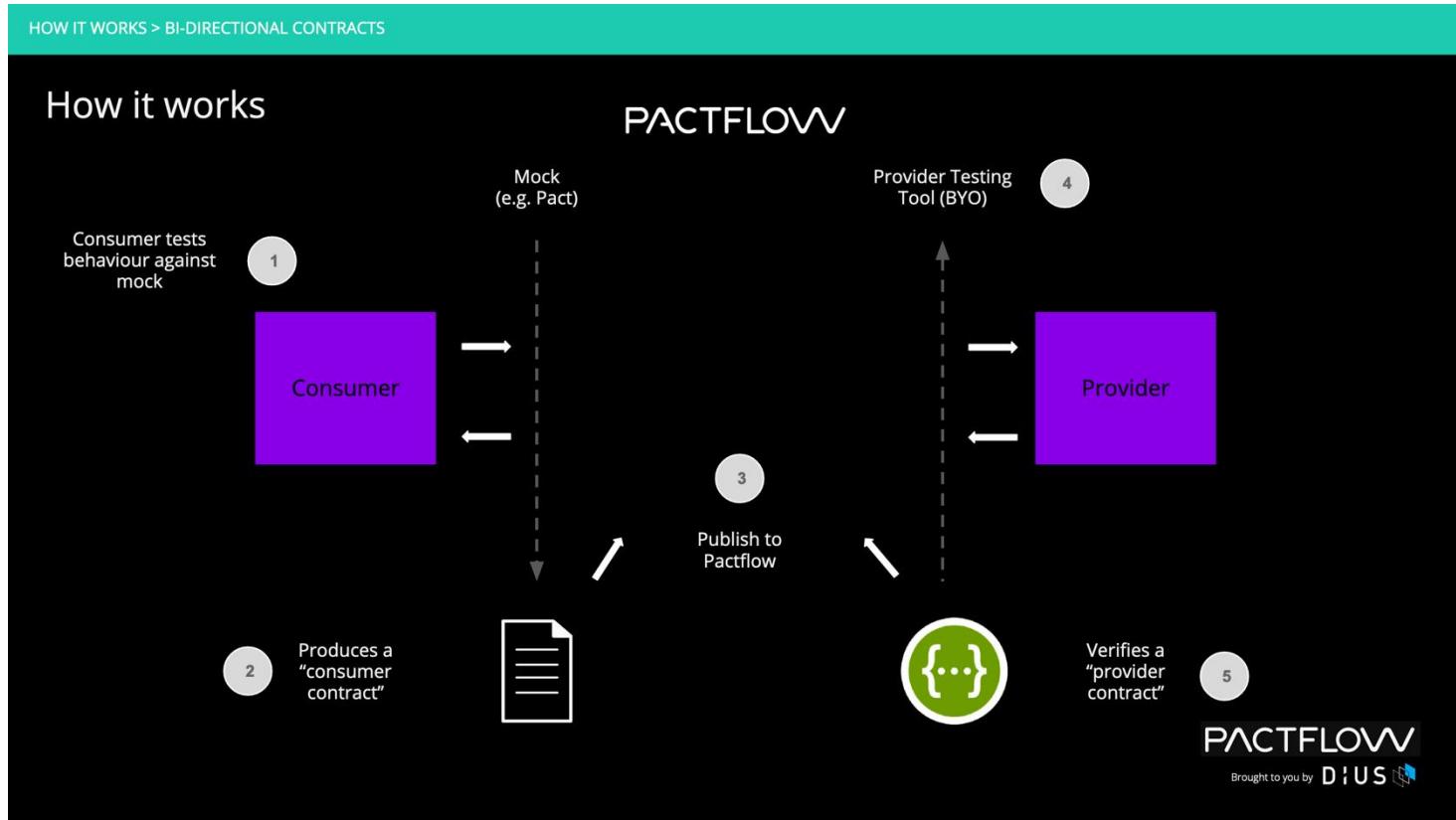
**Version**

**License**

# Pact



<https://pactflow.io/>



# Dredd



<https://dredd.org>

The screenshot shows the official Dredd website at <https://dredd.org>. The header features the Dredd logo (a shield with a red 'M' and yellow stars) and navigation links for 'latest', 'Search docs', 'Edit on GitHub', and 'Dredd – HTTP API Testing Framework'. Below the header, the main content area displays the title 'Dredd – HTTP API Testing Framework' and various status badges: 'npm package 14.1.0', 'FAILED', 'build passing', 'docs passing', 'coverage 92%', 'Snyk security monitored'. A large 'DREDD' logo with the tagline 'No more outdated API Documentation' is prominently displayed. The page also includes a brief description of Dredd's purpose, its language agnosticism, and its validation process. At the bottom, there are sections for 'Features' and 'Supported API Description Formats', along with a sidebar for Algolia AI Search.

**Dredd**

latest

Search docs

Installation Quickstart How It Works How-To Guides Usage JavaScript API Hooks Data Structures Internals

npm package 14.1.0 FAILED build passing docs passing coverage 92% Snyk security monitored

**Dredd – HTTP API Testing Framework**

**DREDD**  
No more outdated API Documentation

Algolia

Dredd is a language-agnostic command-line tool for validating API description document against backend implementation of the API.

Dredd reads your API description and step by step validates whether your API implementation replies with responses as they are described in the documentation.

## Features

## Supported API Description Formats

Find out how Algolia AI Search can instantly and precisely understand your user's intent. [Watch Demo](#)

# Dredd



<https://dredd.org>

The screenshot shows the official Dredd website at <https://dredd.org>. The header features the Dredd logo (a shield with a red 'M' and yellow stars) and navigation links for 'latest' and 'Search docs'. The main content area has a dark sidebar with links like 'Installation', 'Quickstart', 'How It Works', 'How-To Guides', 'Usage', 'JavaScript API', 'Hooks', 'Data Structures', and 'Internals'. The main content area includes a large 'DREDD' logo with the tagline 'No more outdated API Documentation'. It describes Dredd as a language-agnostic command-line tool for validating API descriptions against backend implementations. Below this, there's a section about Dredd's features and supported API description formats. A sidebar on the right contains a snippet of Algolia AI search code.

**Dredd – HTTP API Testing Framework**

npm package 14.1.0 FAILED build passing docs passing coverage 92% Snyk security monitored

**DREDD**  
No more outdated API Documentation

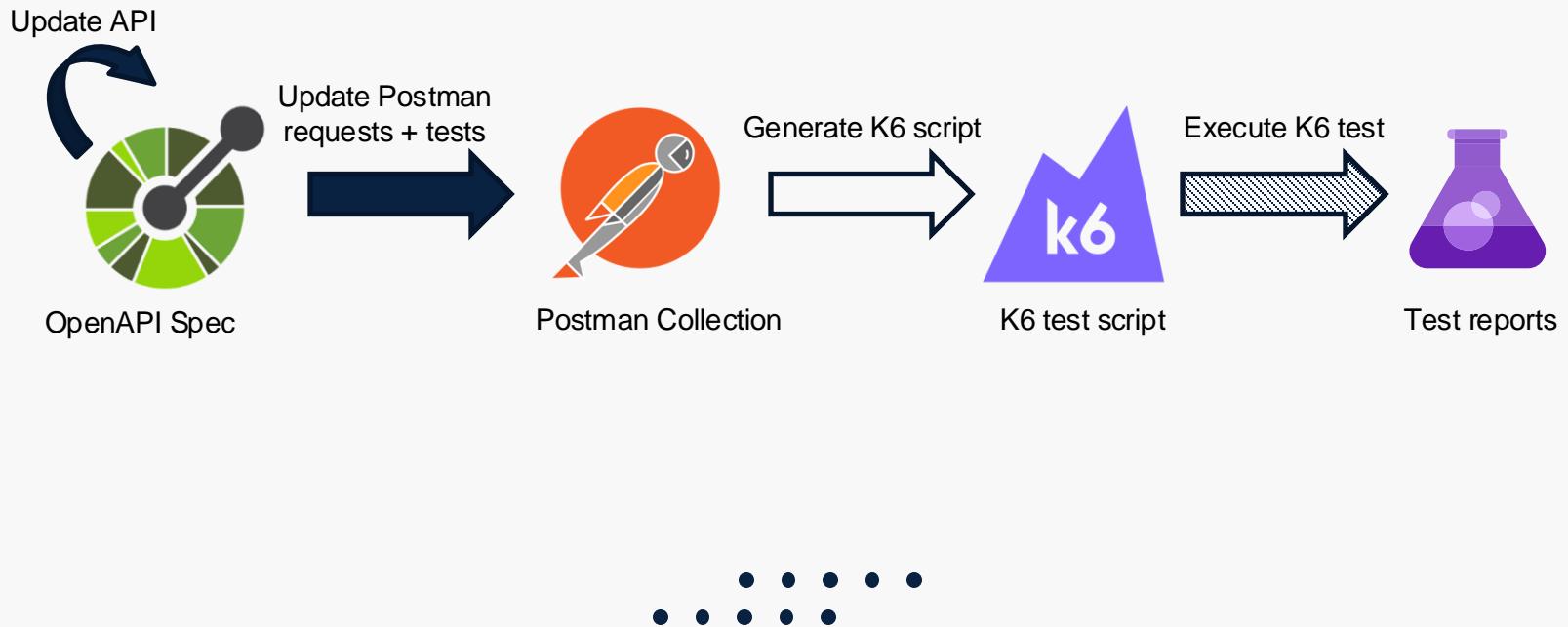
Dredd is a language-agnostic command-line tool for validating API description document against backend implementation of the API.

Dredd reads your API description and step by step validates whether your API implementation replies with responses as they are described in the documentation.

## Features

## Supported API Description Formats

# Performance testing



# K6



<https://k6.io/>

The K6 logo is located at the top left. The navigation bar includes links for OPEN SOURCE, GRAFANA CLOUD K6, PRICING, DOCUMENTATION, ABOUT, SIGN IN, and SIGN UP.

## The best developer experience for load testing

Open source and SaaS for engineering teams

[OPEN SOURCE >](#)

[GRAFANA CLOUD K6 >](#)



LOVED BY DEVELOPERS. TRUSTED BY BUSINESSES



# Gatling



<https://gatling.io/>

The screenshot shows the Gatling website homepage with a dark blue background. At the top left is the Gatling logo (an orange stylized 'G' icon followed by the word 'Gatling'). At the top right is a navigation bar with links: Why Gatling? ▾, Learn more ▾, About ▾, Pricing, Book a demo, Login, and a prominent orange 'Free Trial' button. A large, semi-transparent overlay box is centered on the page, containing the Gatling logo, a red play button icon, and the text 'Test. Succeed. Iterate.' Below this box, the text 'Trusted by 100,000+ businesses worldwide' is displayed. At the bottom left, there is a small orange button labeled 'Let's try it!'. The background features a subtle network graph pattern.

Trusted by 100,000+ businesses worldwide



<https://jmeter.apache.org/>



## About

- [Overview](#)
- [License](#)

## Download

- [Download Releases](#)
- [Release Notes](#)

## Documentation

- [Get Started](#)
- [User Manual](#)
- [Best Practices](#)
- [Component Reference](#)
- [Functions Reference](#)
- [Properties Reference](#)
- [Change History](#)
- [Javadocs](#)
- [JMeter Wiki](#)
- [FAQ \(Wiki\)](#)

## Tutorials

- [Distributed Testing](#)
- [Recording Tests](#)
- [JUnit Sampler](#)

# Apache JMeter™



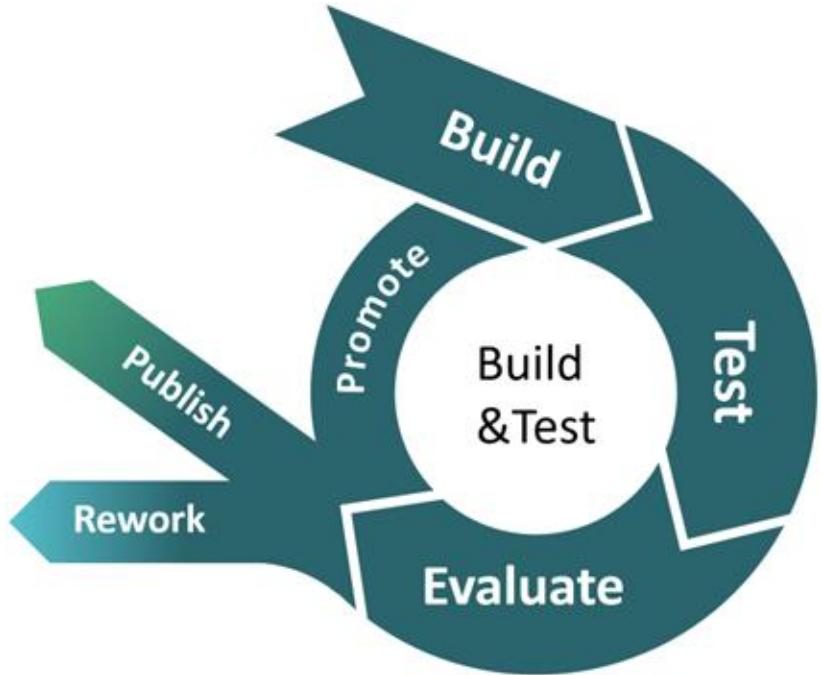
The **Apache JMeter™** application is open source software, a 100% pure Java application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

## What can I do with it?

Apache JMeter may be used to test performance both on static and dynamic resources, Web dynamic applications. It can be used to simulate a heavy load on a server, group of servers, network or object to test its strength or to analyze overall performance under different load types.

Apache JMeter features include:

- Ability to load and performance test many different applications/server/protocol types:
  - Web - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, ...)
  - SOAP / REST Webservices
  - FTP
  - Database via JDBC
  - LDAP
  - Message-oriented middleware (MOM) via JMS
  - Mail - SMTP(S), POP3(S) and IMAP(S)
  - Native commands or shell scripts
  - TCP
  - Java Objects



• • • • •

 #smc-list-api.21067.02\_merge Merge pull request 12945 from story/134362\_links\_to\_records into main  
on ES - smc.list.api - Deploy

Run new

⋮

Summary Tests Environments Associated pipelines

### Summary

2 Run(s) Completed ( 1 Passed, 1 Failed ) [1 unique failing test in the last 14 days](#)

**16**

Total tests

+16



15 Passed  
1 Failed  
0 Others



1 Failed tests (+1)  
1 New  
0 Existing

**93.75%**

Pass percentage

↑ 93.7%

**1m**

Run duration ⓘ

↑ +1m

**0**

Tests not reported

 Bug  Link  Flaky

 Test run ⏺

 Column Options

⋮

Filter by test or run name

Tags ⏺ SmcListsNewman ⏺ Owner ⏺ Passed ⏺ ×

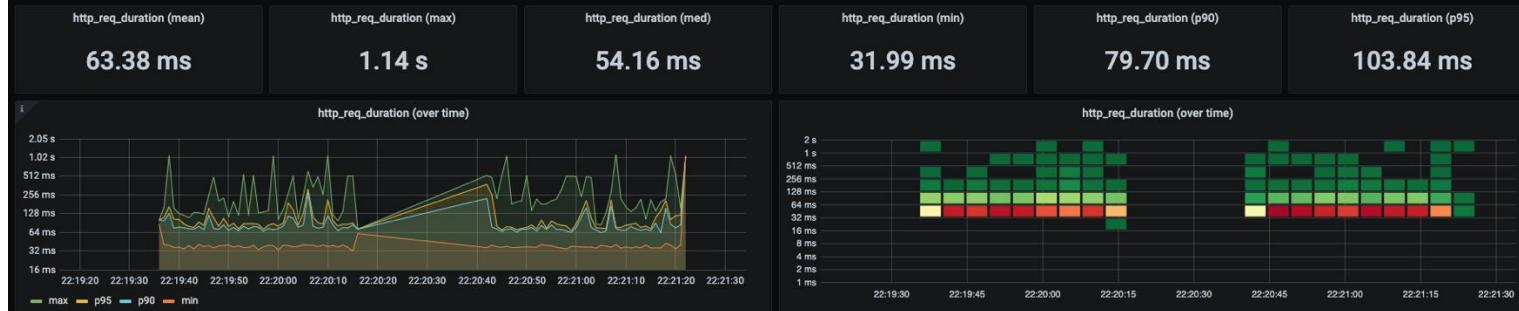
Test	Duration	Failing since	Failing build	Tags
✓ ✅ Newman test suite "List" - buzz-main env (14/14)	0:00:03.194			
✓ [POST] /lists/v2/audiences/{audienceId}/import - Response header Operation-Location is present	0:00:03.196			
✓ [POST] /lists/v2/audiences/{audienceId}/import - Status code is 2xx	0:00:03.194			
✓ [GET] /lists/v2/audiences/{audienceId} - Content-Type is application/json	0:00:01.460			
✓ [GET] /lists/v2/audiences/{audienceId} - Schema is valid	0:00:01.460			
✓ [GET] /lists/v2/audiences/{audienceId} - Status code is 2xx	0:00:01.460			
✓ [GET] /lists/v2/audiences/{audienceId} - Response has JSON Body	0:00:01.457			
✓ [GET] /lists/v2/audiences/{audienceId}/records/{recordId} - Schema is valid	0:00:00.564			
✓ [GET] /lists/v2/audiences/{audienceId}/records/{recordId} - Status code is 2xx	0:00:00.564			

Measurement http\_req\_duration + http\_req\_blocked

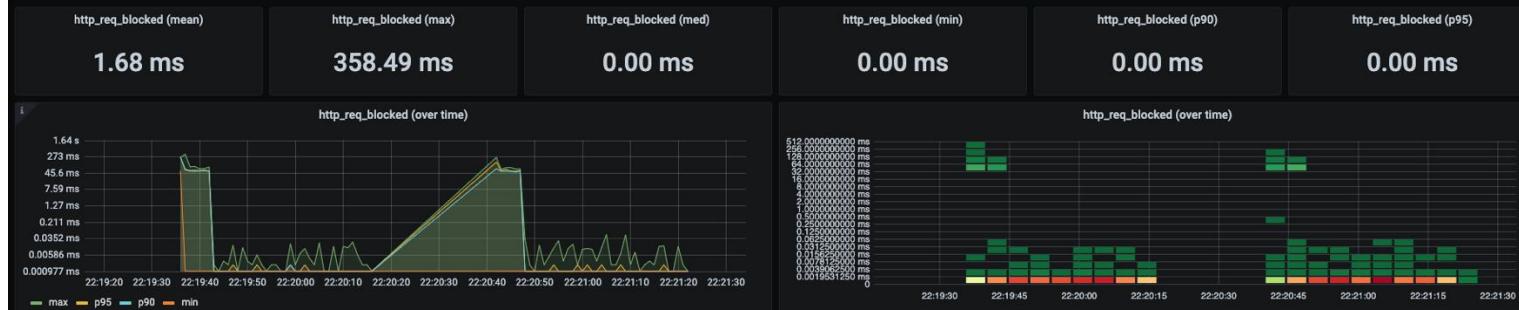
Dashboard Row



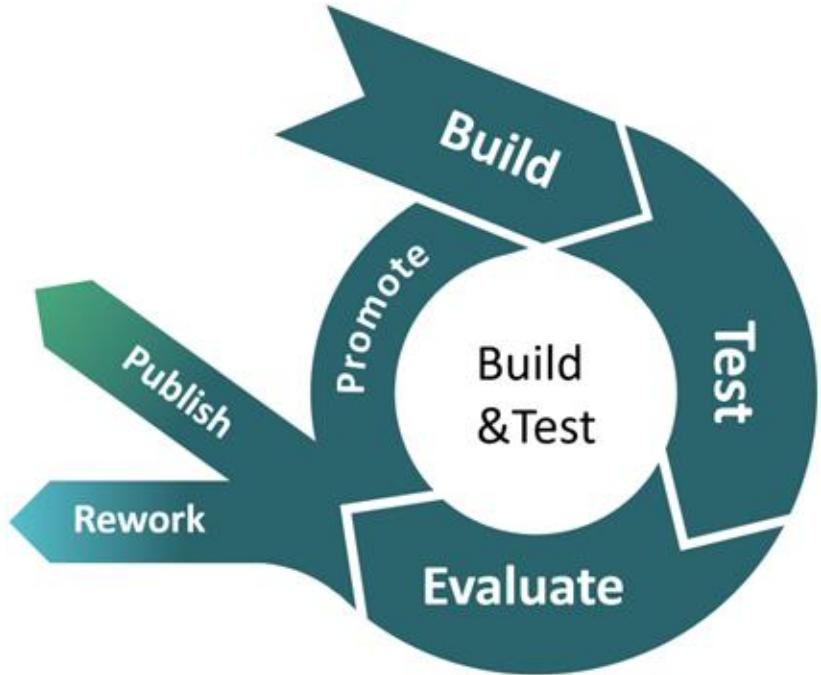
http\_req\_duration



http\_req\_blocked

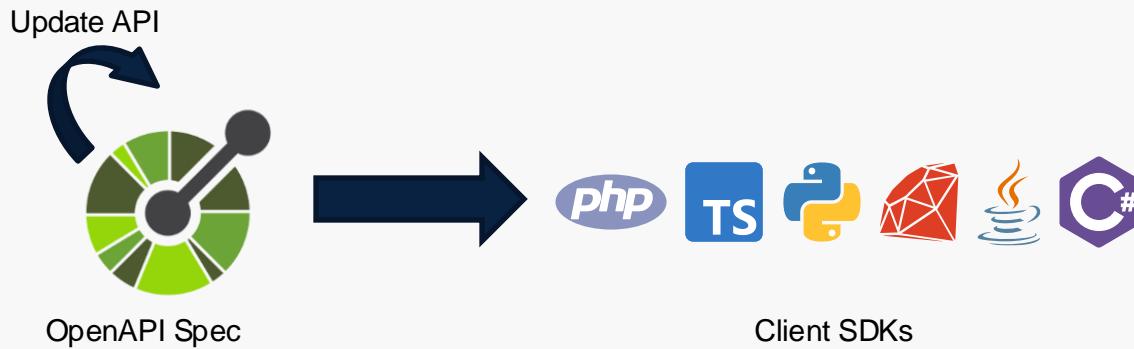


Dashboard Row



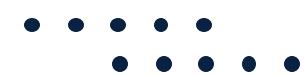
• • • • •

# Generate SDK

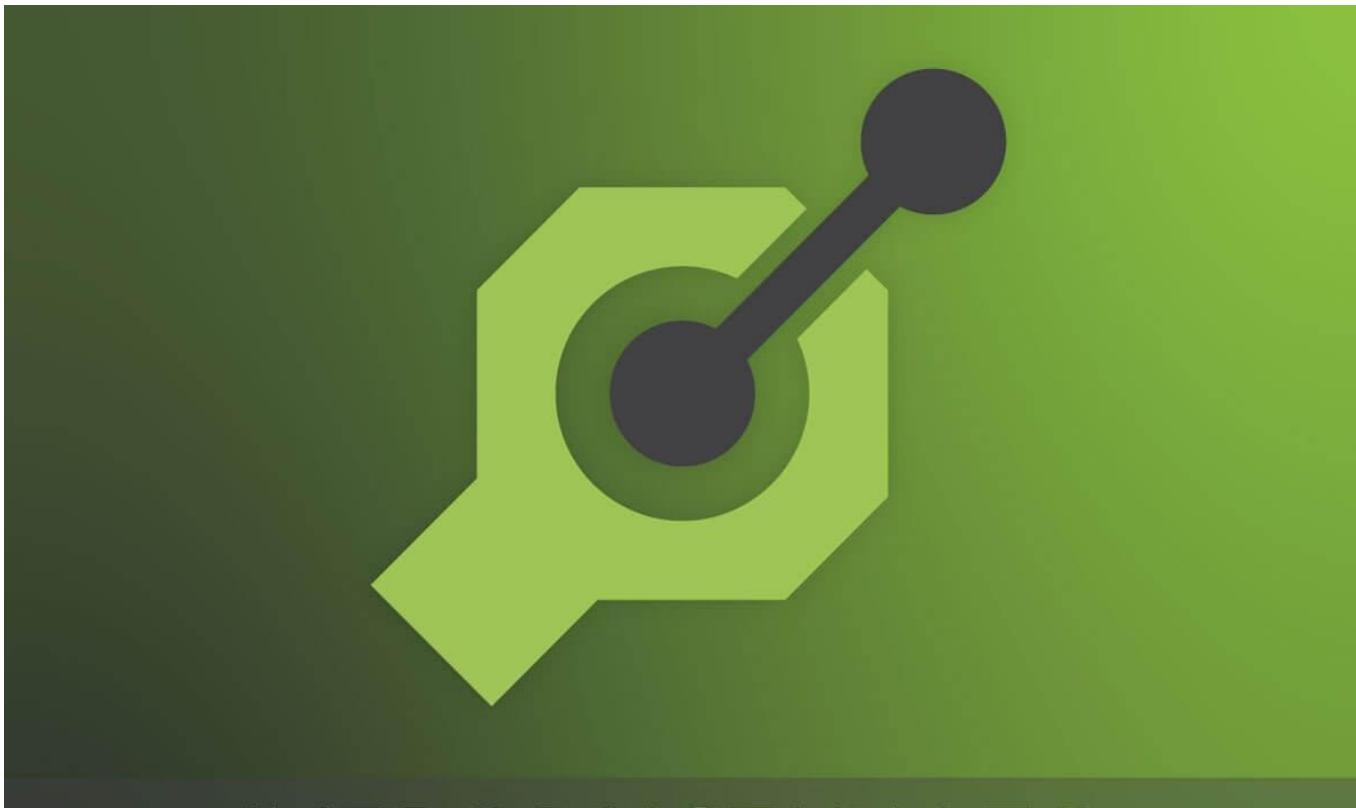


... : : : : :

# SDK Client Generation



<https://openapi-generator.tech>



# SDK Client Generation



<https://orval.dev/>

 **Orval**  
RESTFUL CLIENT GENERATOR

Search docs

Docs   Playground   Examples    

## Generate client with appropriate type-signatures

Generate, validate, cache and mock in your frontend applications all with your OpenAPI specification.

[Get Started](#) [GitHub](#)



**Productivity**  
Save time to drink coffee, not your mind, out of

**Error boundary**  
Get your contract! With the combination of orval

**API mocking**  
Don't wait for your backend to be ready to test

# SDK Client Generation



<https://learn.microsoft.com/en-us/openapi/>

The screenshot shows the Microsoft OpenAPI documentation landing page. It features a blue header with the title "Microsoft OpenAPI documentation" and a sub-header "Learn how to work with OpenAPI-described APIs using Microsoft tools." Below the header are three main navigation cards:

- OVERVIEW** What is Kiota? (Icon: document with a gear)
- TRAINING** Kiota docs (Icon: computer monitor with a book)
- GET STARTED** Get Kiota (Icon: battery with a lightning bolt)

## Kiota

Kiota builds API clients from OpenAPI descriptions

The screenshot shows the Kiota documentation page, which lists the supported programming languages for generating API clients. Each language section includes a logo, the language name, a "Quick start" link, and an "Azure auth tutorial" link.

<b>.NET</b> Quick start Azure auth tutorial	<b>CLI</b> Quick start Azure auth tutorial	<b>Go</b> Quick start Azure auth tutorial	<b>Java</b> Quick start Azure auth tutorial
<b>PHP</b> Quick start Azure auth tutorial	<b>Python</b> Quick start Azure auth tutorial	<b>Ruby</b> Azure auth tutorial	<b>TypeScript</b> Quick start Azure auth tutorial

# SDK Client Generation



<https://buildwithfern.com/>

The screenshot shows the fern website homepage at the top, featuring the fern logo, navigation links (Products, Showcase, Documentation, Pricing, Blog), and a "Get started" button. Below the header is a large call-to-action section with the text "Input your OpenAPI. Output SDKs and Docs." and "Beautiful SDKs and Docs for your API". It includes two buttons: "Book a demo" and "Get started - free →". Below this is a dark-themed screenshot of the Vellum API documentation interface. The interface shows the "List Documents" endpoint under the "API Reference" section. It includes a search bar, a sidebar with navigation links like Help Center, Changelog, and API Reference, and detailed documentation for the "List Documents" endpoint, including query parameters (document\_index\_id, limit, offset) and a code snippet for Node.js. To the right of the main content is a "RESPONSE" panel showing JSON data. On the far right, there's a sidebar titled "Trusted by" listing various companies: Flatfile, MERGE, vellum, AssemblyAI, candidhealth, and Primer.

Input your OpenAPI. Output SDKs and Docs.

Beautiful SDKs and Docs for your API

Book a demo   Get started - free →

vellum 2.3 -

Products   Blog   Free trial

API Reference

List Documents

This endpoint lists all documents. You may optionally filter using query parameters.

Query Parameters

document\_index\_id optional string

Filter down to only those documents that are included in the specified index. You may provide either the Vellum-generated ID or the unique name of the index specified upon initial creation.

limit optional string

Number of results to return per page.

offset optional string

REQUEST (GET)

```
import { VellumClient } from "vellum-ai"
const vellum = VellumClient.api.key("API_KEY")
result = await vellum.documents.list({
  documentIndexId: "6289e235",
  limit: 10,
  offset: 0,
  orderings: "label"
});
```

RESPONSE

```
{
  "count": 10,
  "next": "https://api.vellum.ai/next?page=2",
  "previous": null,
  "results": [
    {
      "id": "6289e235-88c6-25e-97be-cdb419fddd52",
      "external_Id": "doc_e65ce15c",
      "last_uploaded_at": "2023-09-27T00:03:22Z",
      ...
    }
  ]
}
```

Trusted by

Flatfile

MERGE

vellum

AssemblyAI

candidhealth

Primer

# SDK Client Generation



[https://www.speakeeasyapi.dev/](https://www.speakeasyapi.dev/)

The screenshot shows the Speakeasy API homepage with a dark background and yellow accents. At the top, there's a navigation bar with links for Docs, Pricing, Showcase, Product, Changelog, Company, a yellow 'TRY IT NOW' button, and Log in. Below the navigation, the word 'SPEAKEASY' is written in yellow. The main headline reads 'Your API: Enterprise ready in a click' in large white font. Below it, a sub-headline says 'Everything you need to create great integration experiences for your APIs: from native-language SDKs and Terraform providers, to friction-free docs.' At the bottom, there are two buttons: a yellow 'TRY IT NOW →' button and a grey 'BOOK A DEMO' button. On the left side, there's a small 'TRUSTED BY' section with a yellow icon. To the right of the main text area, there are two large, stylized yellow 3D bars.

SPEAKEASY

Docs Pricing Showcase Product Changelog Company TRY IT NOW Log in

## Your API: Enterprise ready in a click

Everything you need to create great integration experiences for your APIs: from native-language SDKs and Terraform providers, to friction-free docs.

TRY IT NOW →

BOOK A DEMO

TRUSTED BY



• • • • •

# Pipeline

Update API



OpenAPI Spec

Update API

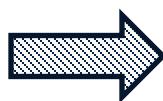


Validate OpenAPI

Build API



k6



Testing



Publish Docs

...



Marvel Universe

DETAILS

STATS

LOGS

ACTIONS

CONFIG

MEMBERS

BACKUPS

DEVELOPMENT

LAUNCHED



SCENARIOS

CUSTOM TASKS

HISTORY

SCHEDULE

Filter by name 

+ CREATE NEW SCENARIO

## Scenarios

### DEPLOY MARVEL API

EDIT

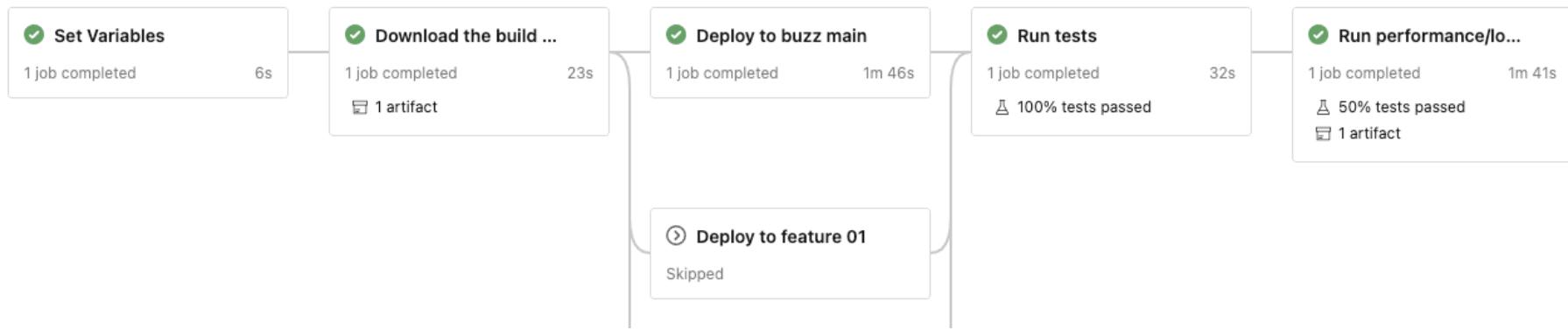


HOOKS

RUN

1. Git Clone with symlink
2. Marvel - Validate OpenAPI spec [Node.js 18] # icon" data-bbox="878 498 898 528"/> 
6. Marvel - Start API [Node.js 18] # icon" data-bbox="878 558 898 588"/> 
7. Marvel - Publish OpenAPI [Node.js 18] # icon" data-bbox="878 618 898 648"/> 
8. Marvel - Publish Postman [Node.js 18] # icon" data-bbox="878 678 898 708"/> 
9. Marvel - Run Contract test [Node.js 18] # icon" data-bbox="878 738 898 768"/> 
10. Marvel - Run Performance test [Node.js 18] # icon" data-bbox="878 808 898 838"/> 
11. Marvel - Run Spike Performance test with Influx reporting [Node.js 18] # icon" data-bbox="878 868 898 898"/> 

Stages Jobs



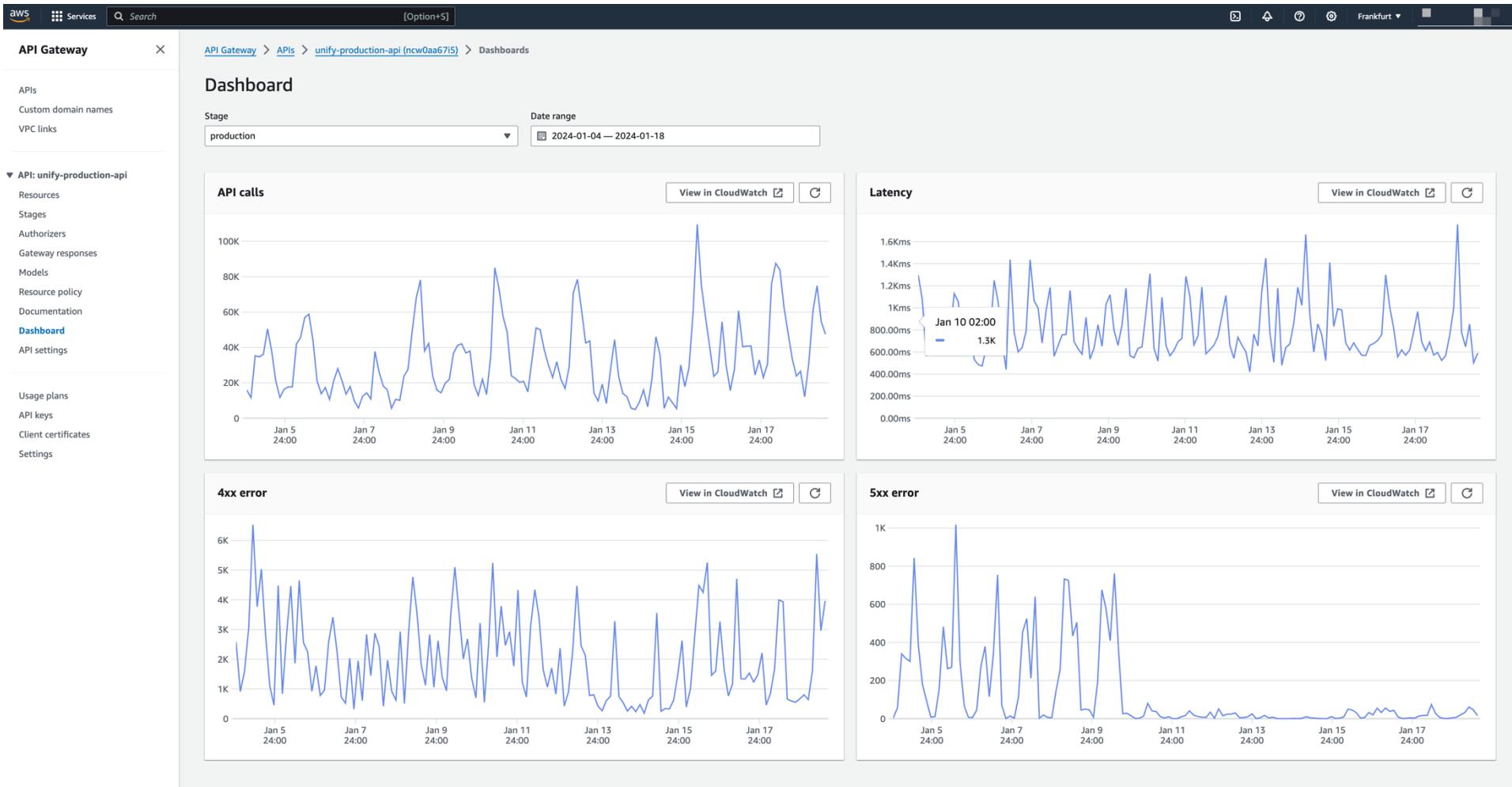
## ✓ Newman :: Filter lists endpoints for Newman test suite



View raw log



```
1 Starting: Newman :: Filter lists endpoints for Newman test suite
2 =====
3 Task      : Bash
4 Description : Run a Bash script on macOS, Linux, or Windows
5 Version   : 3.182.0
6 Author    : Microsoft Corporation
7 Help      : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/bash
8 =====
9 Generating script.
10 Script contents:
11 npm run filter:openapi-newman:lists
12 ===== Starting Command Output =====
13 /usr/bin/bash --noprofile --norc /home/vsts/work/_temp/9cc3a35a-4e51-463d-81a0-95b920d562bb.sh
14
15 > selligent-smc-api@1.0.0 filter:openapi-newman:lists /home/vsts/work/1/s/api-smc
16 > openapi-format ./docs/openapi/smc-rest-v2.openapi.json --configFile configs/openapi-format-config-newman-lists.json
17
18
19 | (index) | Values
20 |-----|
21 | sortFile | 'configs/openapi-format-sort.json'
22 | sort     | true
23 | lineWidth | -1
24 | configFile | 'configs/openapi-format-config-newman-lists.json'
25 | verbose  | true
26 | output    | 'testing/newman/smc-rest-v2-newman-lists.openapi.json'
27 | filterFile | 'configs/openapi-filter-options-newman-lists.json'
28 | rename    | 'SMC-NEWMAN-lists'
29
30 Sort File: configs/openapi-format-sort.json
31 Filter File: configs/openapi-filter-options-newman-lists.json
32 Input file: ./docs/openapi/smc-rest-v2.openapi.json
33 OpenAPI title renamed to: "SMC-NEWMAN-lists"
34 Output file: testing/newman/smc-rest-v2-newman-lists.openapi.json
35
36 ✓ OpenAPI was formatted successfully
37 Finishing: Newman :: Filter lists endpoints for Newman test suite
```



Last 24 hours Roles = All

Server Browser

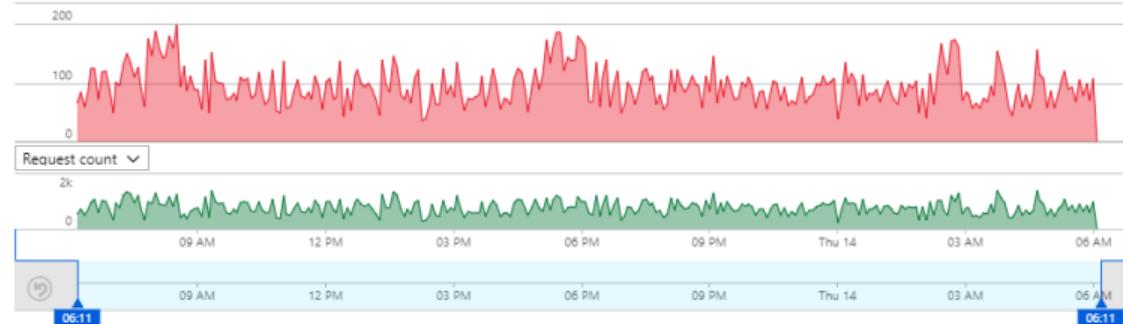
Operations Dependencies Exceptions Roles

View in Analytics

Feedback

Refresh

Failed request count



Select operation

Search to filter items...

OPERATION NAME

USERS COUNT (FAILED) COUNT PIN

Overall	5.26k	27.53k	219.54k
GET ServiceTickets/Details	1.39k	13.86k	13.87k
GET Employees/Details	1.38k	4.63k	9.24k
POST ServiceTickets/Create	1.36k	4.41k	4.41k
GET Employees/Create	874	2.90k	2.90k
GET Home/Index	839	873	9.91k
GET ServiceTickets/Index	404	404	14.26k

Overall

Top 3 response codes

COUNT	FILTERING
500	24.61k
404	2.92k

Top 3 exception types

COUNT	FILTERING
NullReferenceExce...	18.49k
SqlException	4.41k
HttpException	2.92k

Top 3 dependency failures

COUNT	FILTERING
SQL	12.17k
Azure blob	9.04k

Drill into...

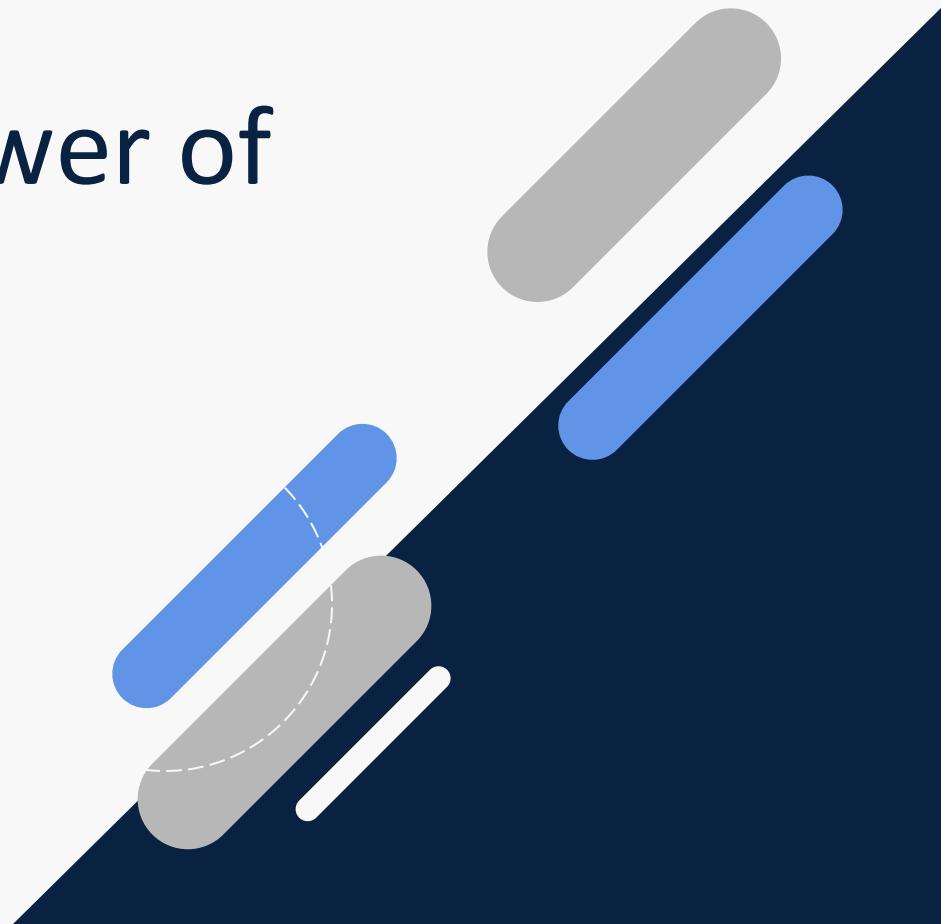
27.53k Operations

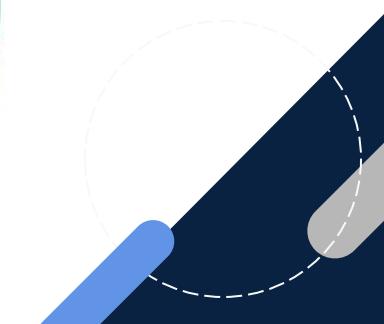
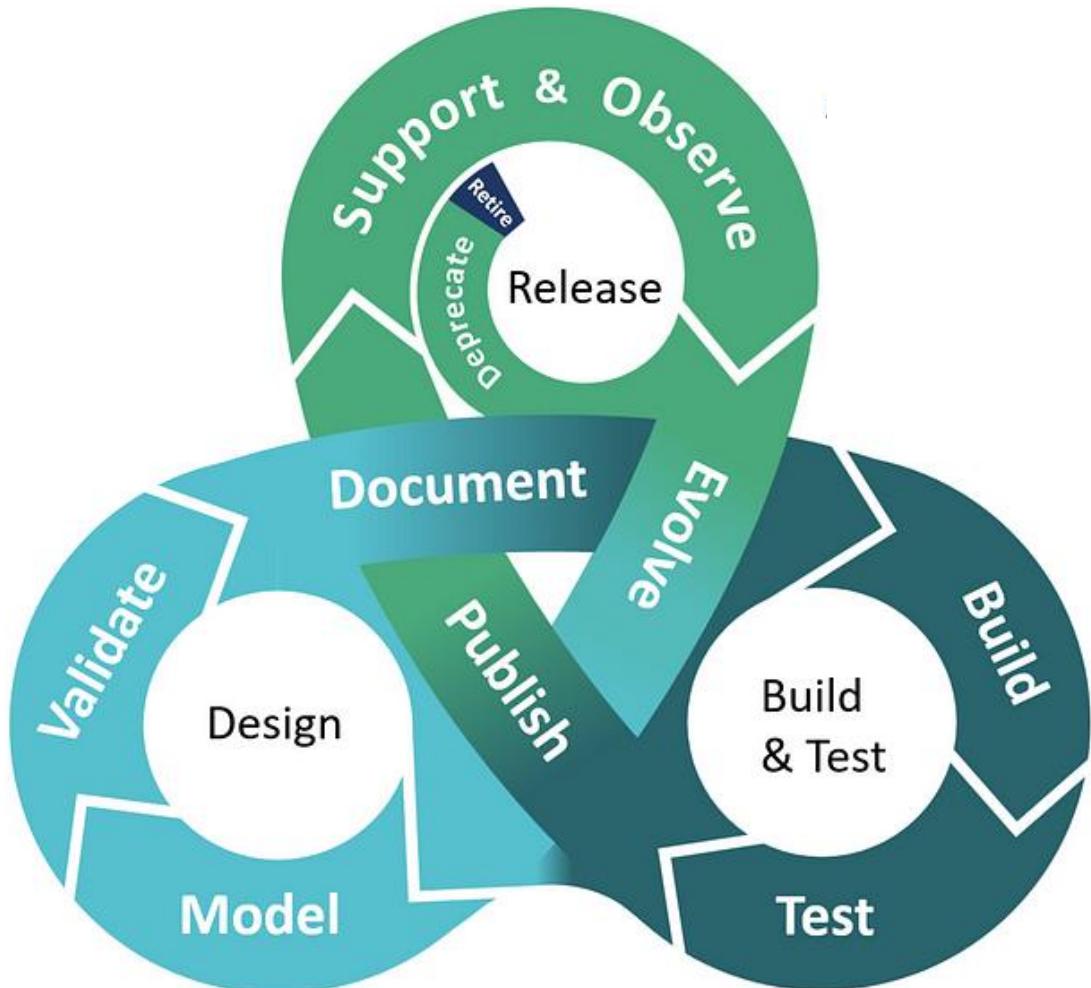


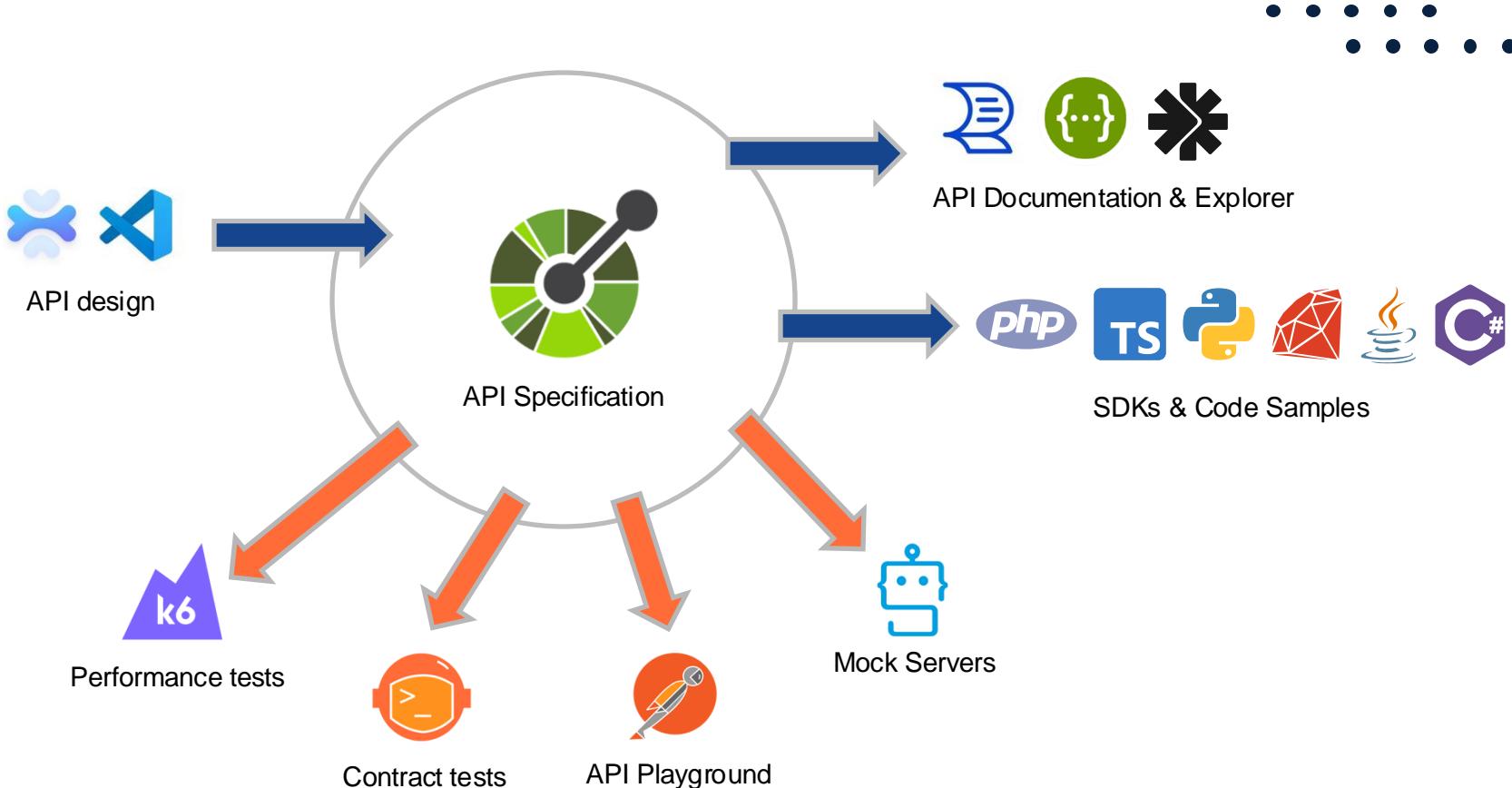
# Unlocking the Power of Spec-Driven API Development

Recap

...:::







# Spec-driven benefits

- Better **documentation**
  - Clearer **requirements**
  - Shorter **feedback loops**
  - More **consistent API design**
  - Easier Team **alignment**
  - Docs that **match** the API (and vice versa)
  - Much **lower** updating effort
  - Automatable outputs like:
    - **Documentation**
    - **Tests**
    - **Playgrounds**
    - **SDKs**
- • • • •

# Thanks!

Do you have any questions?

