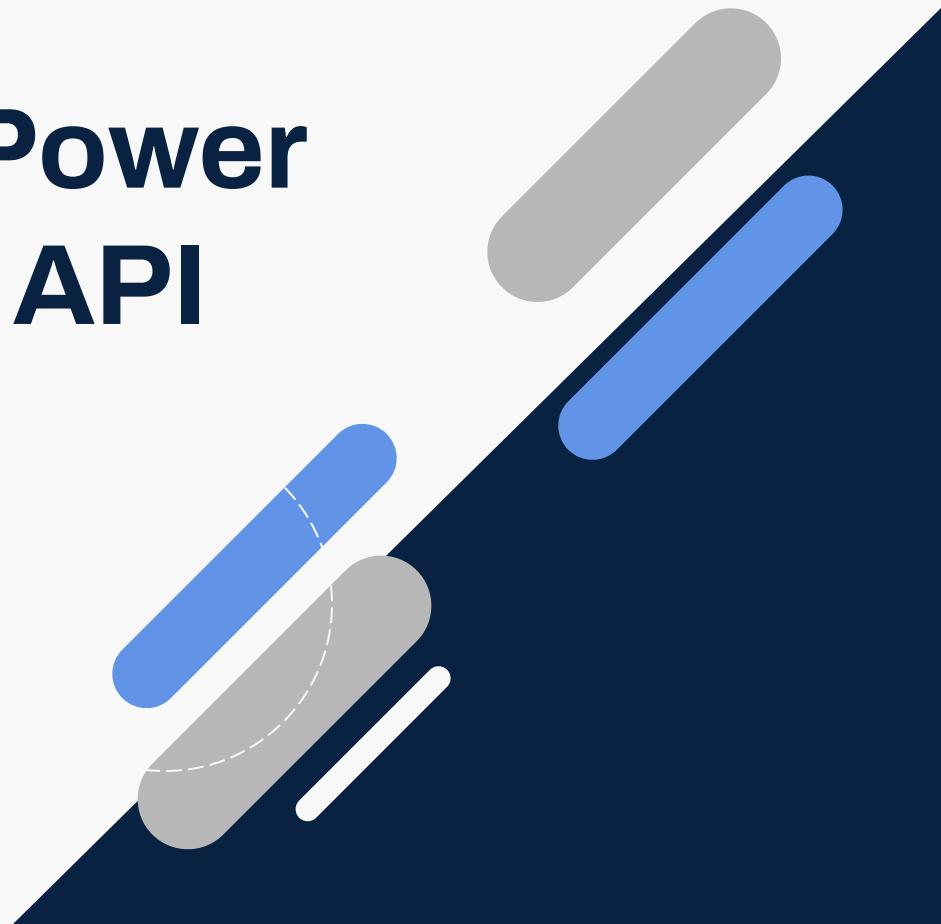


# Unlocking the Power of Spec-Driven API Development

Meetup - Hasselt

Jan 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?



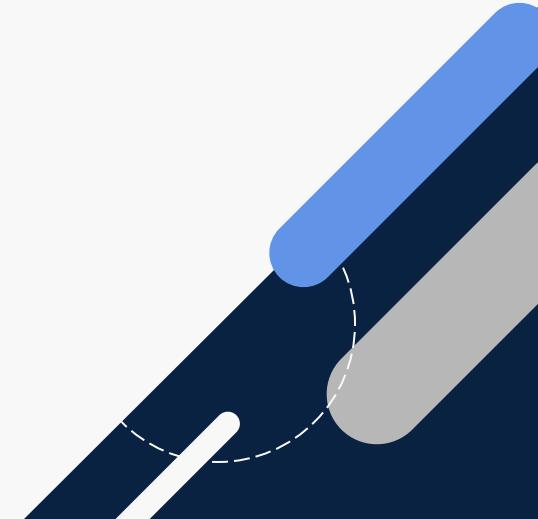
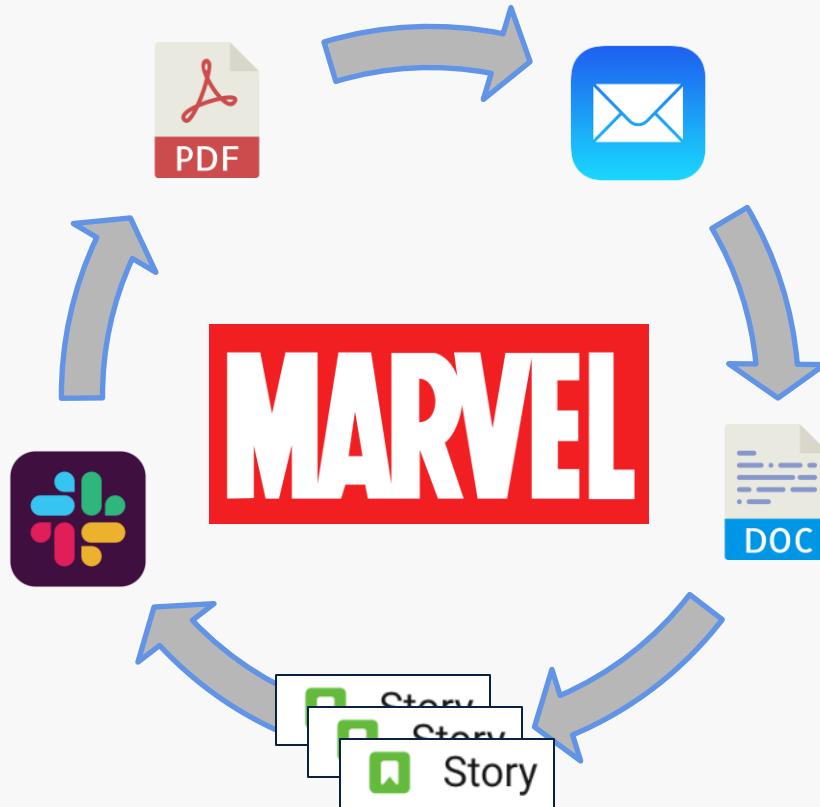
# Which fields are required?



You got a feature request  
Start all over again!



# Let's quickly create, some ...



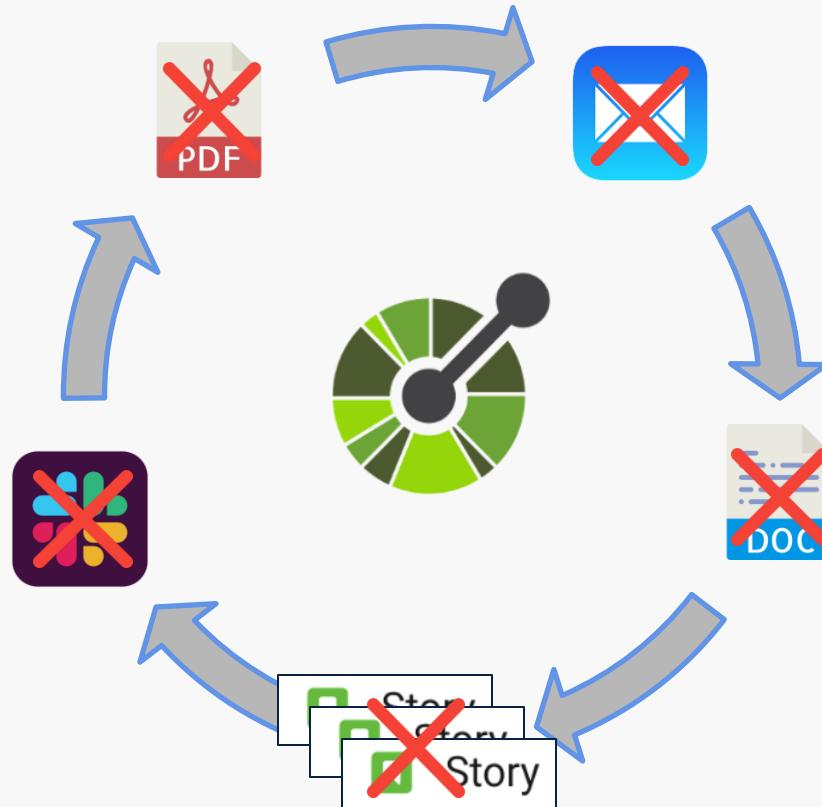
# You are not alone API docs are in poor condition

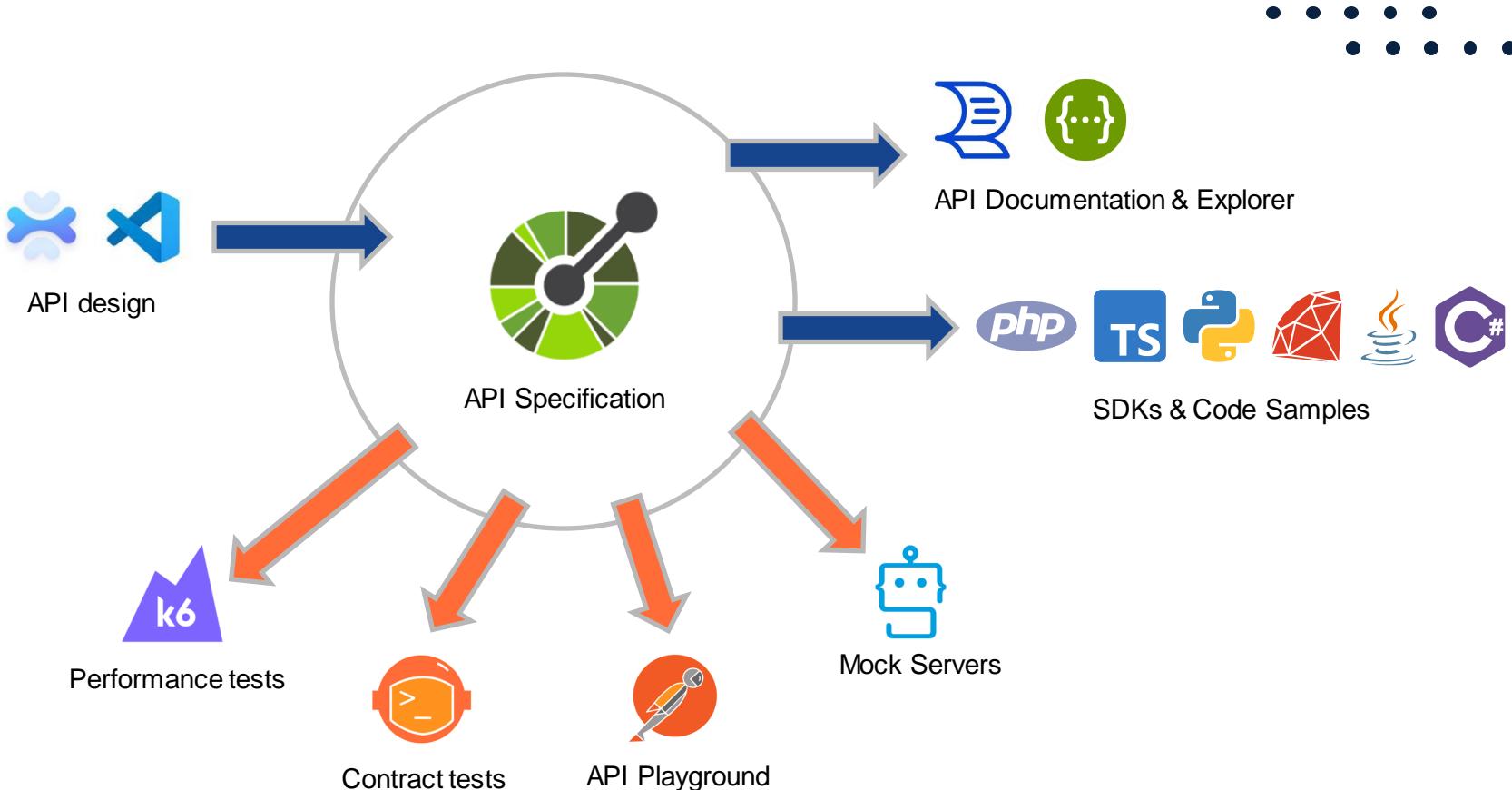


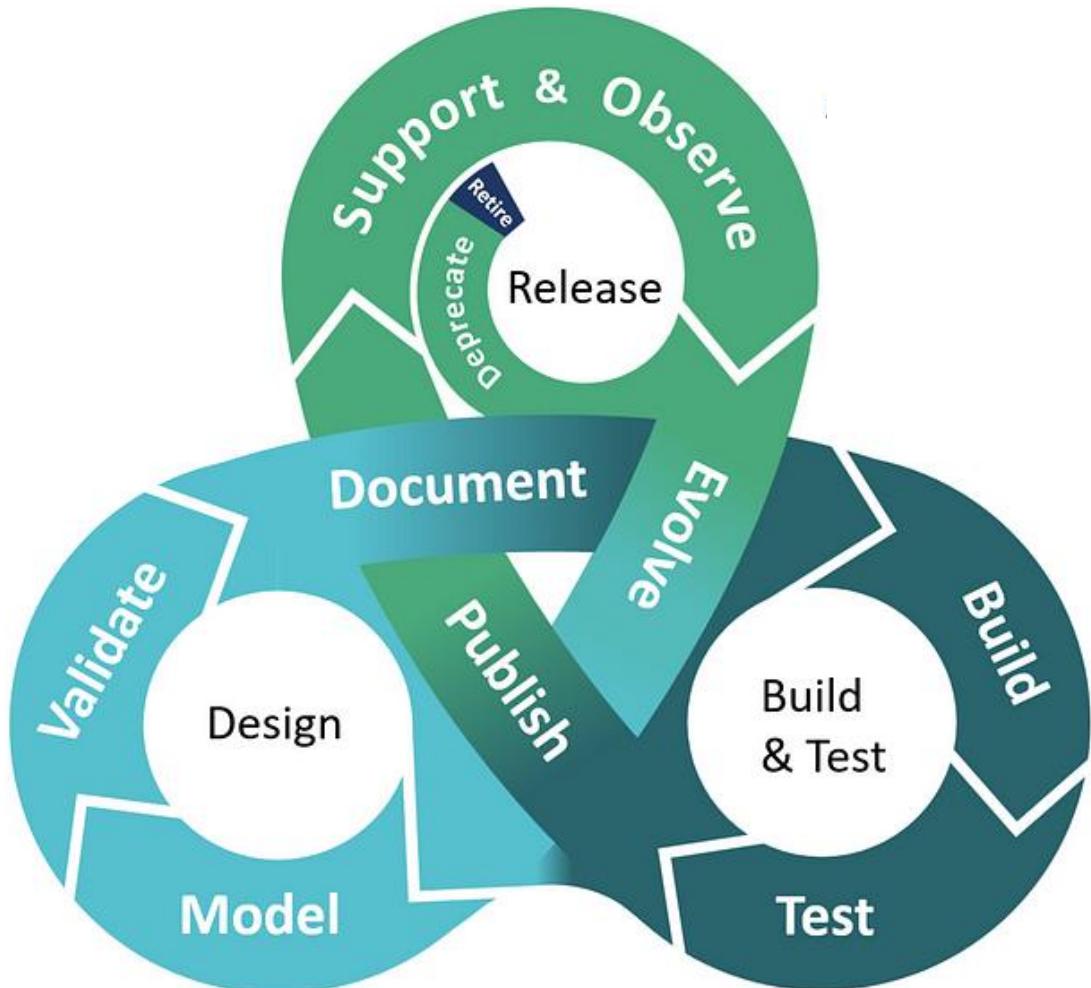
# Let's fix this



# 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**



RAML



api blueprint

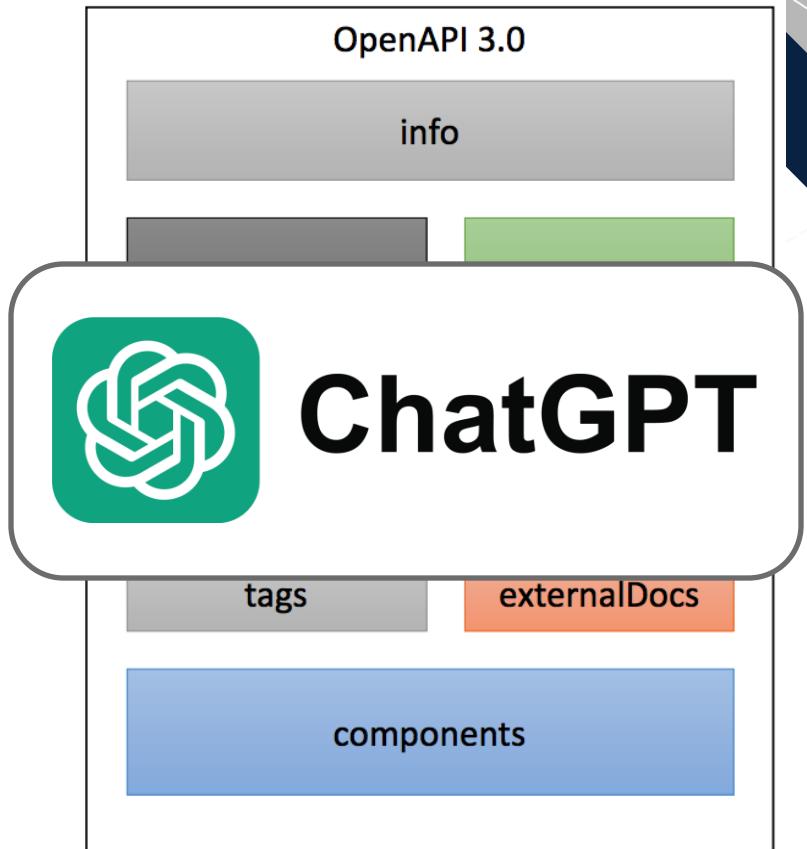
WADL

• • • • •



- 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

• • • • •





# OPENAPI 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

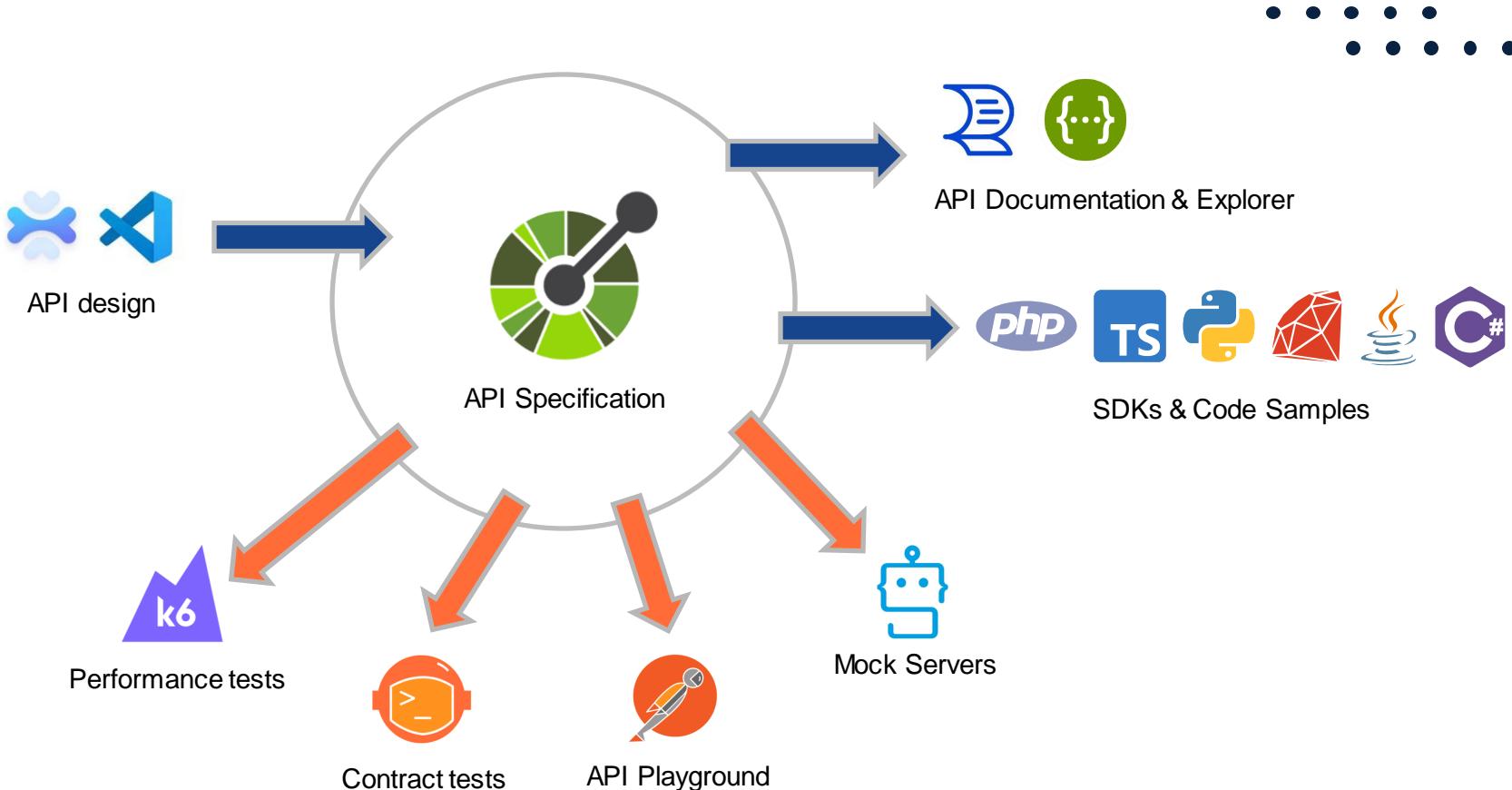
 **APIs-guru/openapi-directory**  
Wikipedia for Web APIs. Directory of REST API definitions in OpenAPI 2.0/3.x format  
[api](#) [aws](#) [azure](#) [rest-api](#) [opendata](#)  
3.5k · Updated 17 hours ago

 **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.



**CODE-FIRST**



Business Input



Write Code



Document



**DESIGN-FIRST**



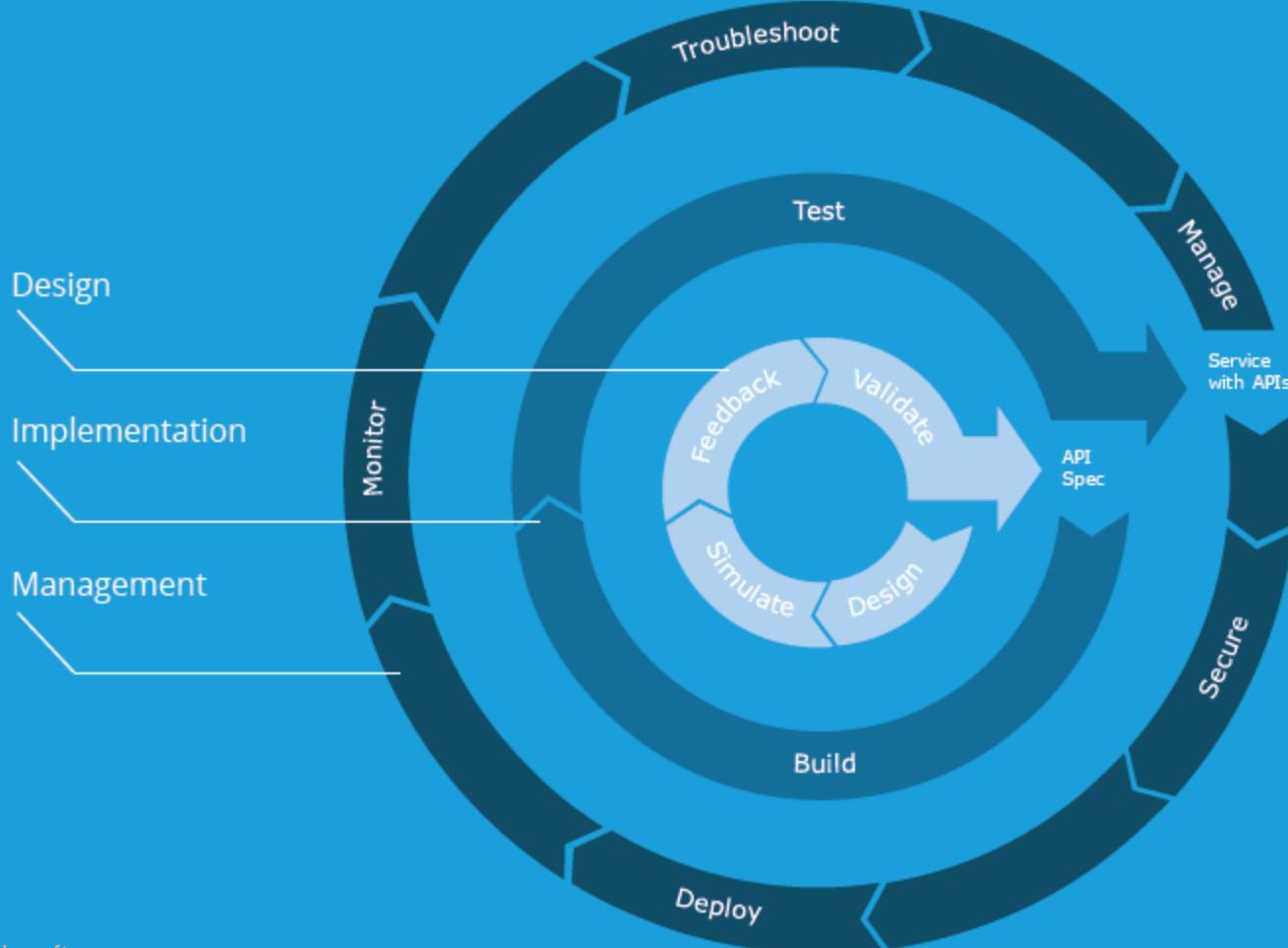
Business Input

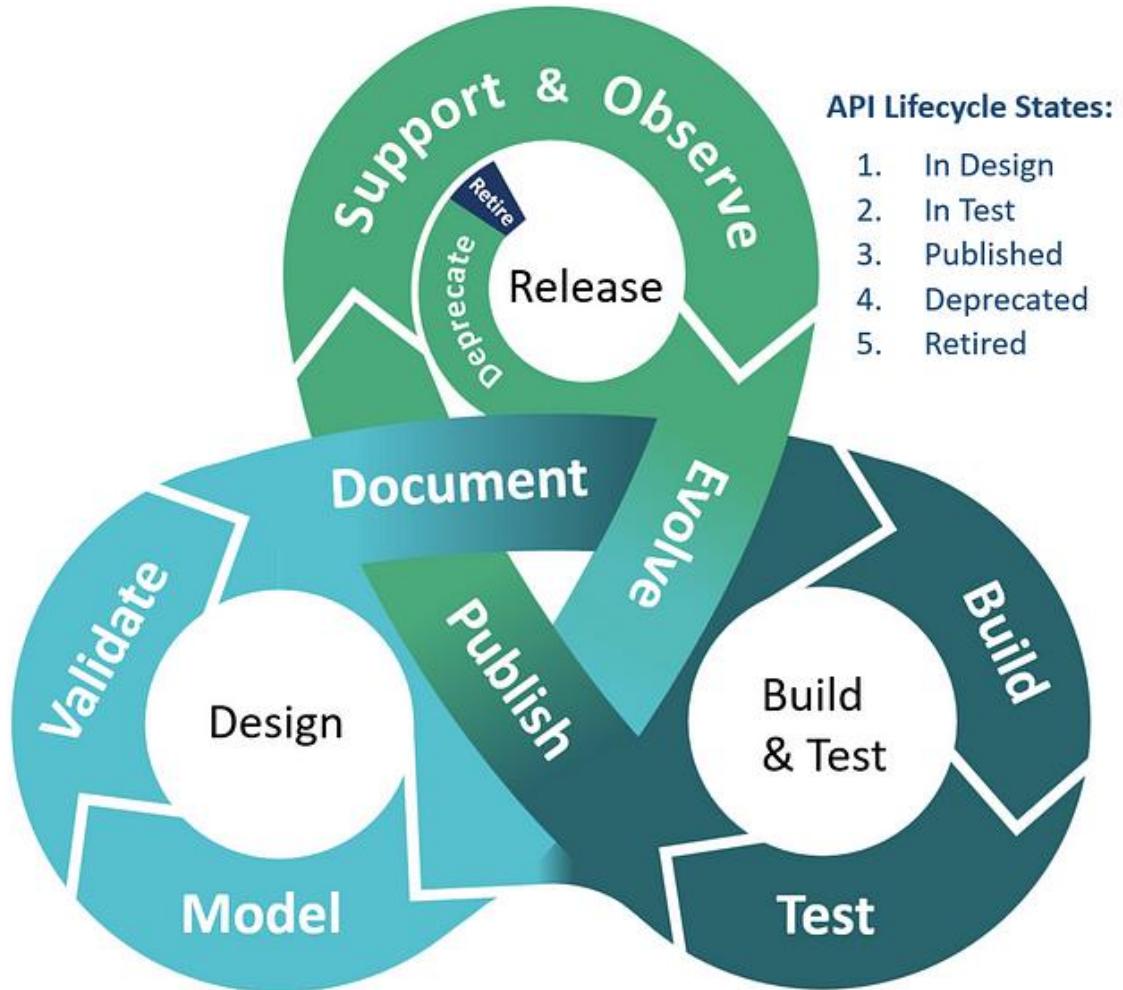


Design  
+ Document



Write Code



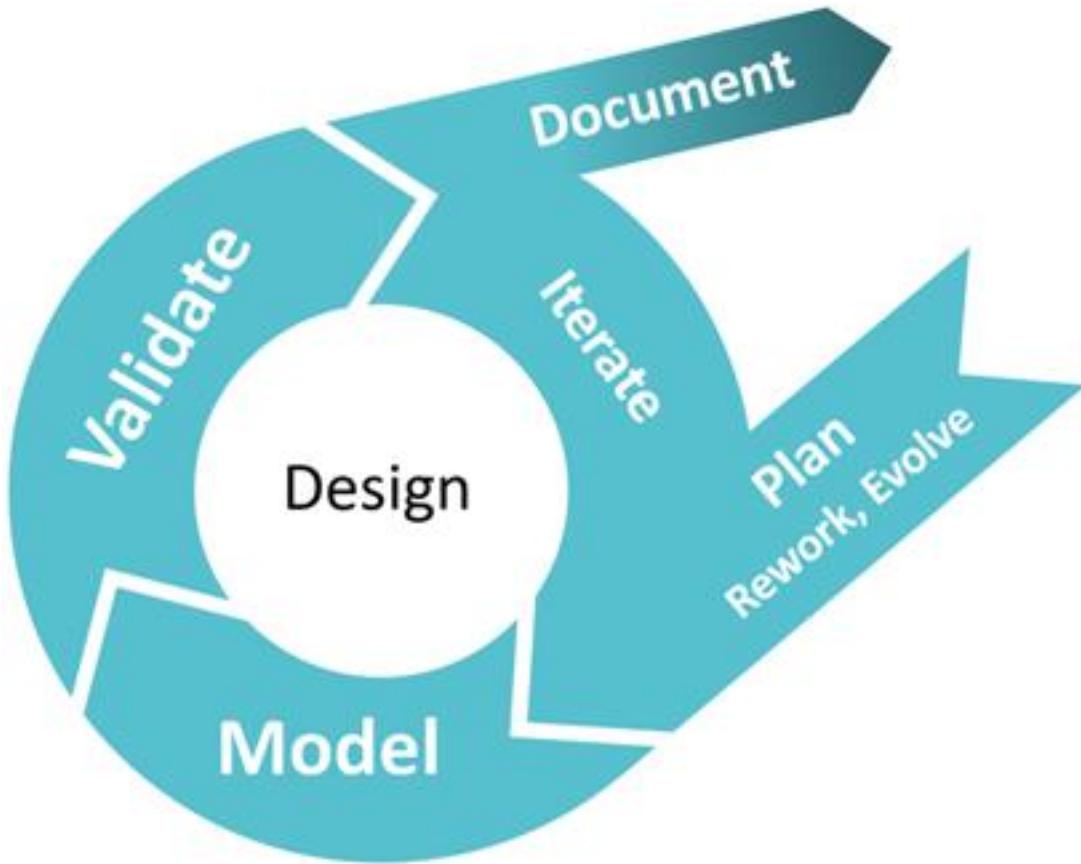


API Lifecycle States:

1. In Design
2. In Test
3. Published
4. Deprecated
5. Retired

# 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 interface with the "openapi.yml" file open in the main editor window. The file contains the OpenAPI specification for a Marvel API. The code is color-coded for syntax highlighting, and the editor features a vertical navigation bar on the left and various toolbars at the top and bottom.

```
marvel-api - openapi.yml
Project  openapi.yml
  Project  marvel-api - /Sites/marvel-api
    > docs-redocly
    > mock-prism
    > publish-openapi
    > publish-postman
    restapi-server
      prisma
      public
      src
        .env
        .gitignore
        package.json
        package-lock.json
      samples
      sdk
      testing-contract
        testing-k6
          config
          k6
        report
          report.html
          report-graph.html
  Structure  OpenAPI 3.0 openapi.yml
    servers
    > Paths
      > /characters
      > /characters/{id}
      > /teams
      > /teams/{id}
    Components
    > Parameters
    > Schemas
    > Security Schemes
  Document 1/1  components:
```

```
description: Production
paths:
  /characters:
    get:
      operationId: getCharacters
      summary: Get all Marvel characters
      description: Retrieve a list of all Marvel characters.
      parameters:
        - $ref: '#/components/parameters/PageParam'
        - $ref: '#/components/parameters/LimitParam'
      responses:
        '200':
          description: Successful operation
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/CharactersResponse'
        '400':
          description: Bad request
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
          tags:
            - Characters
        post:
          operationId: postCharacters
          summary: Create a new Marvel character
          description: Endpoint to create a new Marvel character.
          requestBody:
            required: true
```

At the bottom of the screen, there are several status icons and text indicating sync settings with Composer and PHP language level information.

# 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" project under a "Sample Project".

**Left Sidebar:**

- APIs:** Overview, APIs (Root, Sample APIs (5), pet (3), store (4), user (8)).
- Testing:**
- Settings:**
- Share:**
- Schemas:**
- Requests:**
- Trash:**

**Top Bar:**

- Home, Sample Project, Select environment (dropdown), User icon.

**Central Area:**

- Overview:** Shows the "GET Find pet by ID" operation.
- API Tab:** Selected. Shows the method **GET** and endpoint **/pet/{petId}**. Buttons: Save, Run, Delete.
- Params:** Tabbed section showing **Params** (selected), Body, Cookies, Headers, Auth, Settings, Pre Processors, Post Processors.
- Query Params:** Name, Type, Example, Description. Example: Add a new param.
- Path Params:** Name, Type, Example, Description. Example: petId (integer \*).
- Responses:** successful operation(200), Invalid ID supplied(400), Pet not found(404), Public Responses 0, + Add.
- HTTP Status Code:** 200, **Name:** successful opera, **Content Type:** JSON.
- Generate from JSON etc.:** A button.
- Model Definition:** Shows the **Pet** model with properties: id (integer<int64>), Mock, Description.

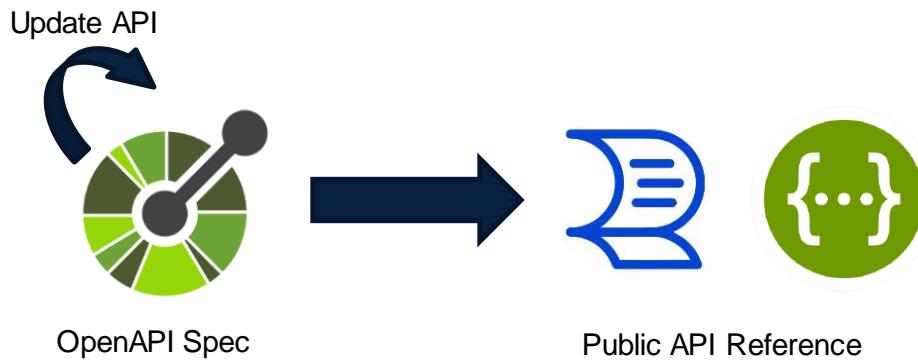
# Stoplight Studio

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



The screenshot shows the Stoplight Studio interface for the Swagger Petstore API. On the left, the sidebar displays the project structure under "studio-demo". The "To-dos" section is expanded, showing the "/todos/{todoid}" endpoint with its methods: GET, PUT, and DELETE. The "GET" method is highlighted. Below this, the "/todos" endpoint is shown with GET and POST methods. The main panel is titled "Get Todo" and shows the endpoint URL `http://todos.stoplight.io /todos/{todoid}`. It lists the available HTTP methods: GET, POST, PUT, PATCH, DELETE, HEAD, OPTIONS, and TRACE. A placeholder for "Endpoint description..." is present. Below the methods are buttons for "+ Security", "+ Header", and "+ Query Param". Under the response section, there are buttons for "+ Response", "200" (green dot), "404" (orange dot), and "500" (red dot). A "200" status code is selected. A "Response description..." field and a "Headers +" button are also visible. At the bottom, there are navigation icons for "my-sug", "remove tags", and "7 Issues".

# 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 Get all audience configurations

200 Successful response

RESPONSE SCHEMA: application/json

GET /lists/audiences

**Response samples**

200 422

Content type application/json

Copy Expand all Collapse all

```
{  
    "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=  
        }  
    ]  
}
```

# 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>

Search ⌘K

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 ↗

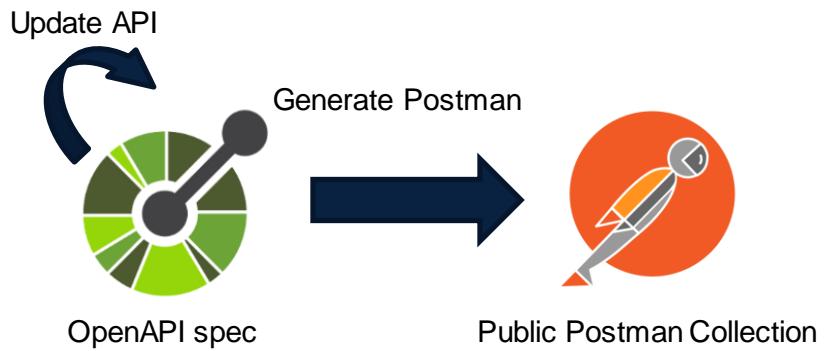
```
1 <?php
2
3 $client = new \GuzzleHttp\Client();
4
5 $response = $client->request('POST', 'http://localhost:3004/
6 'body' => [
7 'first_name' => 'Rocket',
8 'last_name' => 'Raccoon',
9 'name' => 'Rocket Raccoon',
10 'description' => 'A genetically modified raccoon with a knac
11 'powers' => [
12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

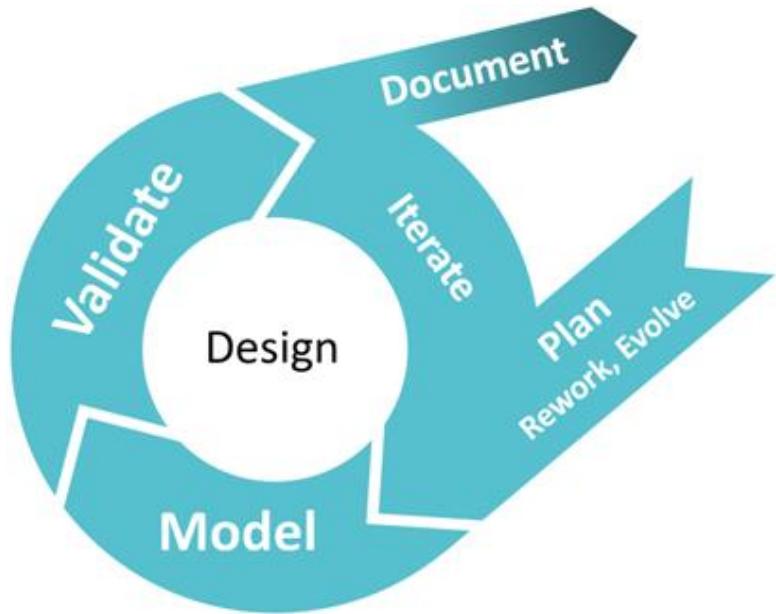
The screenshot shows the Postman application interface. On the left, there's a sidebar with various sections: Collection, Environment, APIs, Mock Servers, Monitors, Flows, and History. The main area displays a collection named "Notion API / Databases / Retrieve a database". Inside this collection, there is one request:

**GET** https://api.notion.com/v1/databases/:id

**Params**:  
Key: id Value: {{DATABASE\_ID}} Description: Required. Enter database id.

**Body**:  
Pretty JSON:  
1 {  
2 "Publisher": {  
3 "id": "%3E%24Pb",  
4 "name": "Publisher",  
5 "type": "select",  
6 "select": {  
7 "options": [  
8 {  
9 "id": "c5ee409a-f307-4176-99ee-6e424fa89afa",  
10 "name": "NYT",  
11 "color": "default"  
12 }  
13 }  
14 }  
15 }  
16 }

On the right side of the interface, there is a "Documentation" panel for the same endpoint. It includes the URL, a brief description ("Retrieves a database object using the ID specified in the request path."), an "Authorization" section ("Bearer Token"), and a "Request Header" section ("Notion-Version 22-02-22"). Below the documentation, there is a "Path Variables" section with the variable "id" and its description ("Required. Enter database id."). At the bottom right, there is a link to "View complete collection documentation →".



• • • • •

# Mock API

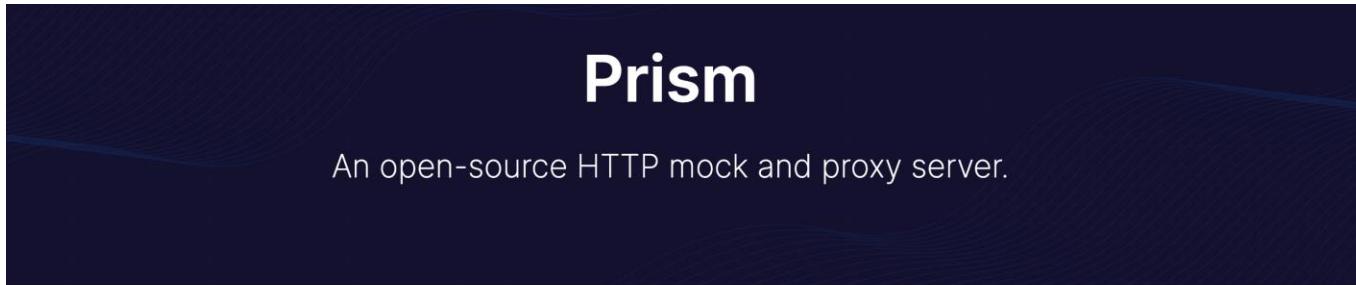


...

# Prism



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



The screenshot shows the Prism landing page. The title "Prism" is at the top center in a large white font. Below it is a subtitle: "An open-source HTTP mock and proxy server." The background is dark blue with light blue wavy patterns.

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

# JSON-Server



<https://github.com/typicode/json-server>

# typicode/json-server



Get a full fake REST API with zero coding in less than 30 seconds (seriously)

72

Contributors

424k

Used by

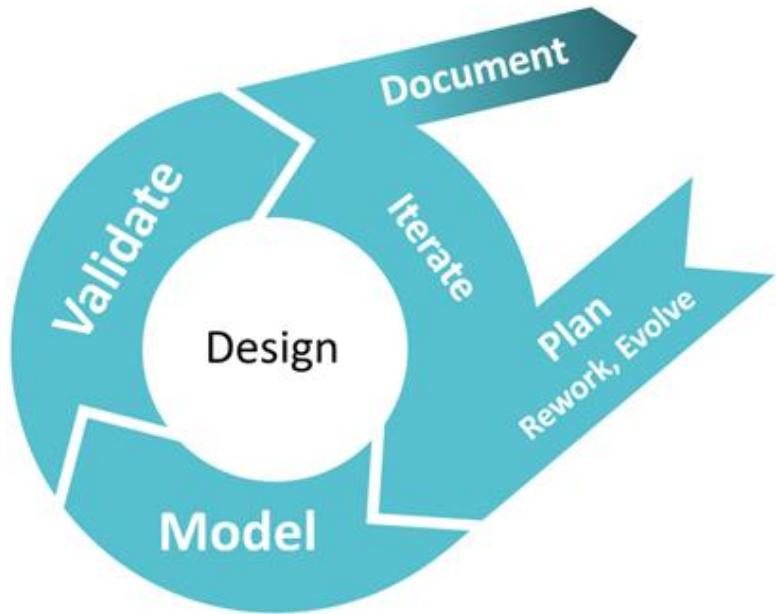
70k

Stars

7k

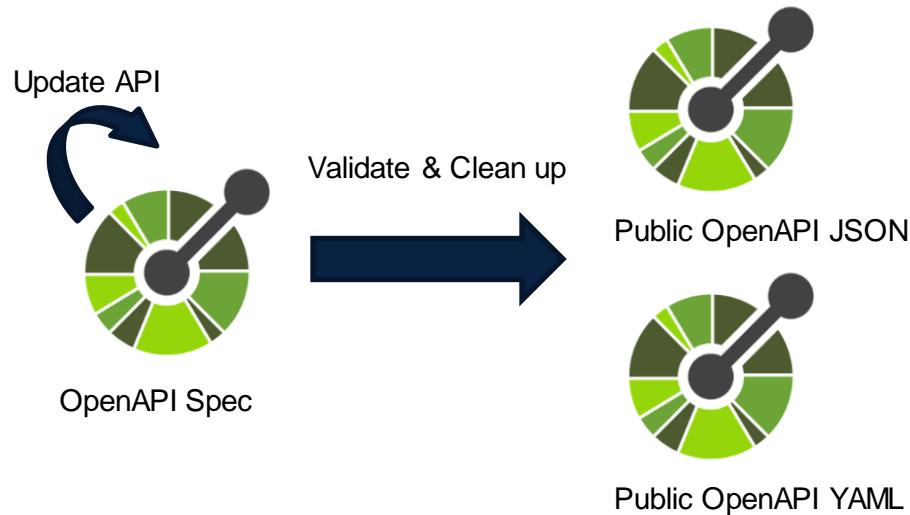
Forks





• • • • •

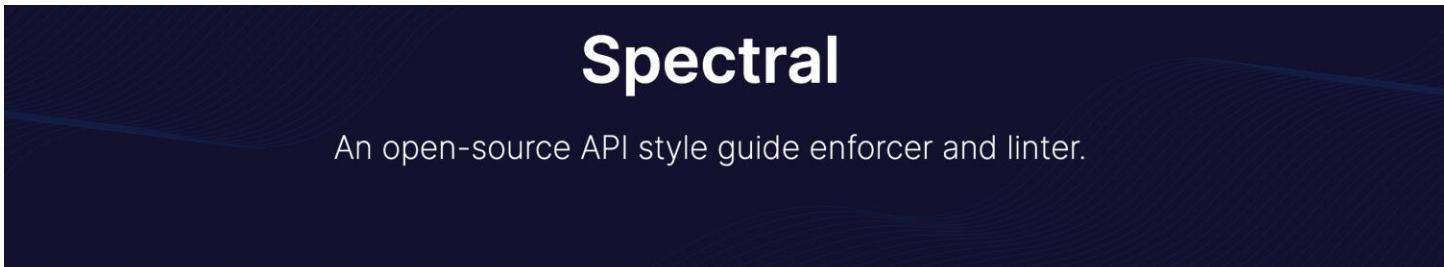
# Validate & Publish OpenAPI



# Spectral



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



The screenshot shows the Spectral landing page. At the top center is the word "Spectral" in a large, white, sans-serif font. Below it is a subtitle: "An open-source API style guide enforcer and linter." The background is dark blue with subtle light blue wavy patterns.

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 interface. 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 is a large white area containing a validation summary for a project named 'main#1'. The summary includes the date 'Thu Aug 19 2021', status 'published', and build time 'Built in: 3s'. It features a 'RESTART' button and a 'GET SNAPSHOT' button. Below this is a horizontal progress bar with four green circles and checkmarks, labeled 'Provision', 'Validate and Build', 'Assets upload', and 'Add to registry'. A green success message at the bottom says 'Woohoo! Your OpenAPI definition is valid.' A 'Results' section at the bottom shows a green checkmark and the text 'No errors or warnings found.' There's also 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.

# 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`

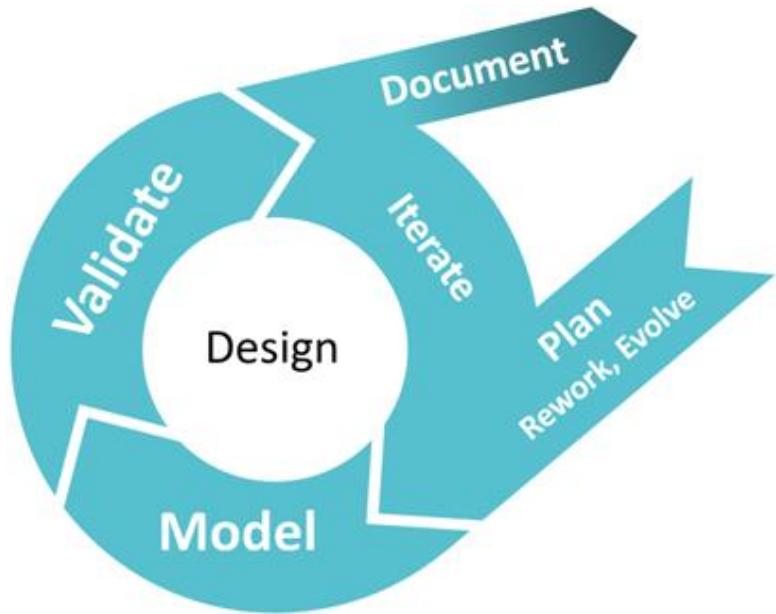
Repository  
[github.com/thim81/openapi-format](#)

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

• • • • •

# Redocly



<https://redocly.com>

The screenshot shows the Redocly homepage with a dark blue background. At the top left is the Redocly logo (a yellow and green icon with the word "Redocly" next to it). At the top right are "Login" and "Sign Up" buttons. The main title "Make API docs your superpower" is displayed prominently in white text. Below the title is a subtitle: "Beautiful API documentation loved by teams and API consumers. Brought to you by the open-source extraordinaires behind Redoc." A large orange "Sign Up" button is located below the subtitle. To the right of the text area is a graphic featuring three overlapping purple cards with icons: a gear, a checkmark, and a document. In the background, there's a blurred image of a computer screen displaying API documentation with a "Try it" button.

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 Readme.com interface for the Hoot API. The top navigation bar includes links for 'Hoot Blog' and 'We're Hiring!'. The main content area is titled 'Send an API Log to Hoot' and contains a 'POST' request example:

```
POST https://metrics.hoot.io
```

The text explains: "Send an API Log to Hoot, so that you can view developer usage metrics about your API. For more information see <https://docs.readme.com/metrics/docs/viewing-api-metrics-in-hoot>".

On the left sidebar, under 'API ENDPOINTS', there are several sections like 'Intro to the Hoot API', 'Authentication', 'Limiting API Results', and 'Developer Metrics API'. Under 'Developer Metrics API', 'API Logs' is expanded, showing 'Send an API Log to Hoot' with a 'POST' button.

The right side of the interface includes sections for 'LANGUAGE' (Shell, Node, Ruby, PHP, Python), 'AUTHENTICATION' (Basic, username:password), and a 'CURL' section with code examples:

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

A 'Try It!' button is located in the CURL section. Below it is a 'RESPONSE' section with the status code '202' and the message 'Finished saving (not necessarily successful)'. A note says 'Updated over 1 year ago'.

At the bottom, there's a feedback poll: 'Did this page help you?' with 'Yes' and 'No' buttons, and a 'Metrics API Introduction' link.

# MkDocs



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

The screenshot shows the Material for MkDocs documentation site. The header has a dark blue background with the title "Material for MkDocs". Below it is a navigation bar with links: Home, Getting started, Setup, Plugins, Reference, Insiders, and Blog. To the right of the navigation is a search bar and a GitHub icon with the repository details: squidfunk/mkdocs-material, 9.5.3, 16.8k stars, and 3.2k forks.

[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

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  
Id of an existing user.

required

#### Responses

200 404

User Found

#### Body

application/json

```
id integer  
Unique identifier for the given user.  
  
firstName string  
  
lastName string  
  
email string<email>  
  
dateOfBirth string<date>
```

required

required

required

required

#### Parameters

userId\* : integer

Send API Request Live Server

#### Request Sample: Shell / cURL

```
curl --request GET \  
--url http://localhost:3000/users/userId \  
--header 'Accept: application/json'
```

#### Response Example

```
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

page

1

The page number for pagination.

limit

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 OK

json

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

[View More](#)

## JUMP TO

### Introduction

#### Characters

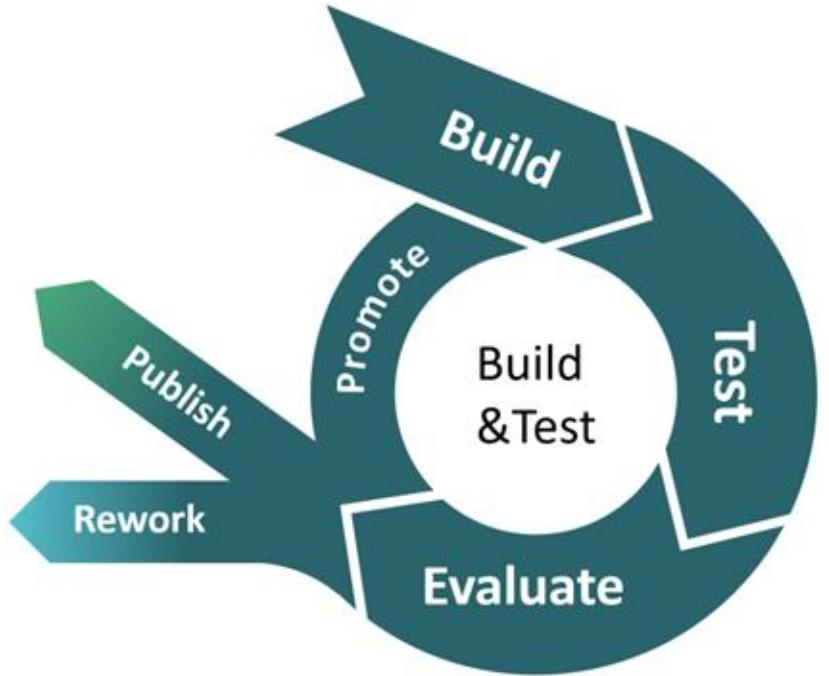
[GET](#) Get all Marvel characters

[POST](#) Create a new Marvel character

[GET](#) Get a Marvel character

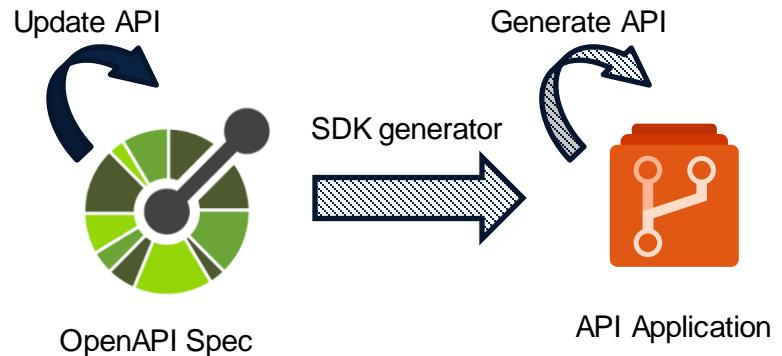
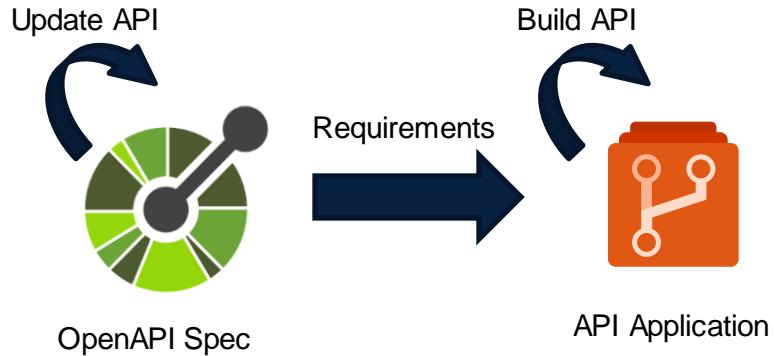
[PUT](#) Update a Marvel character

[DEL](#) Delete a Marvel character



• • • • •

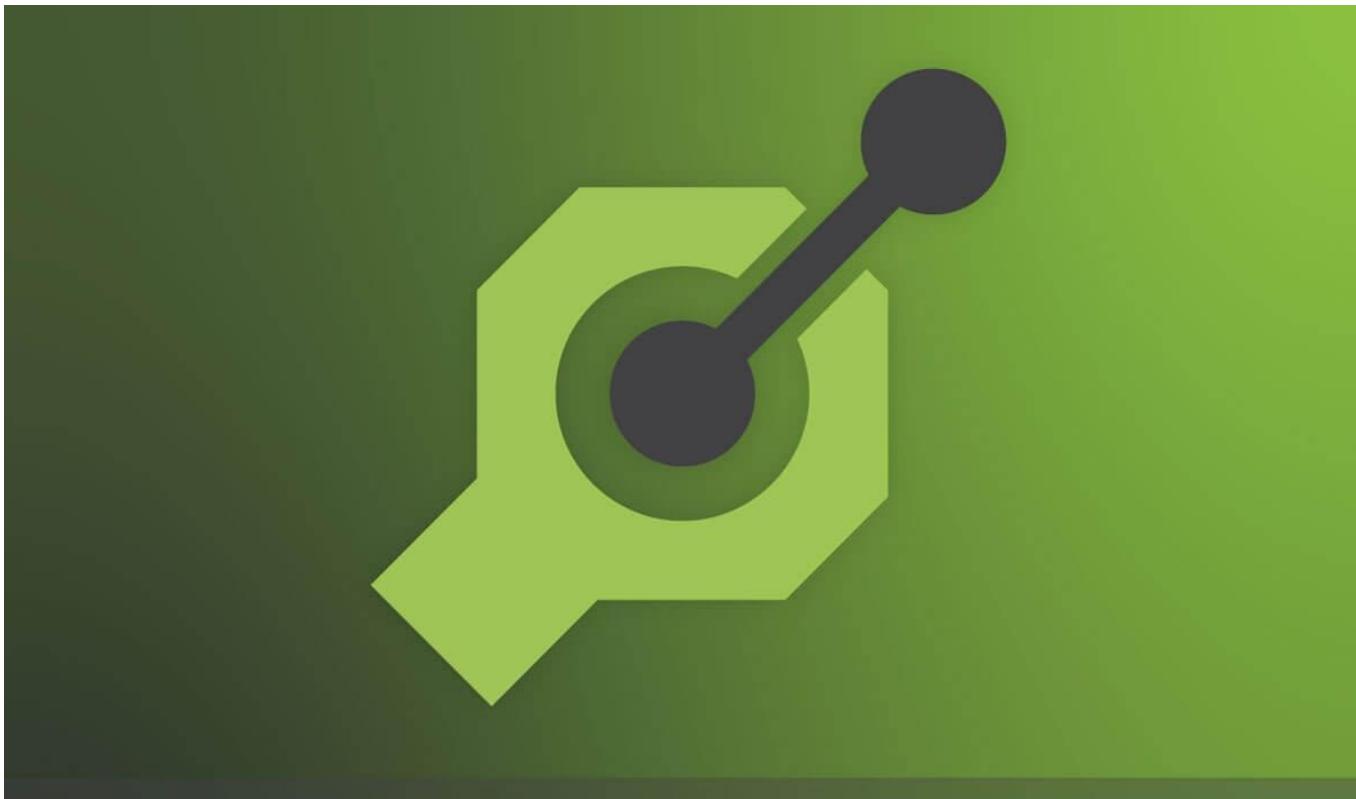
# Build

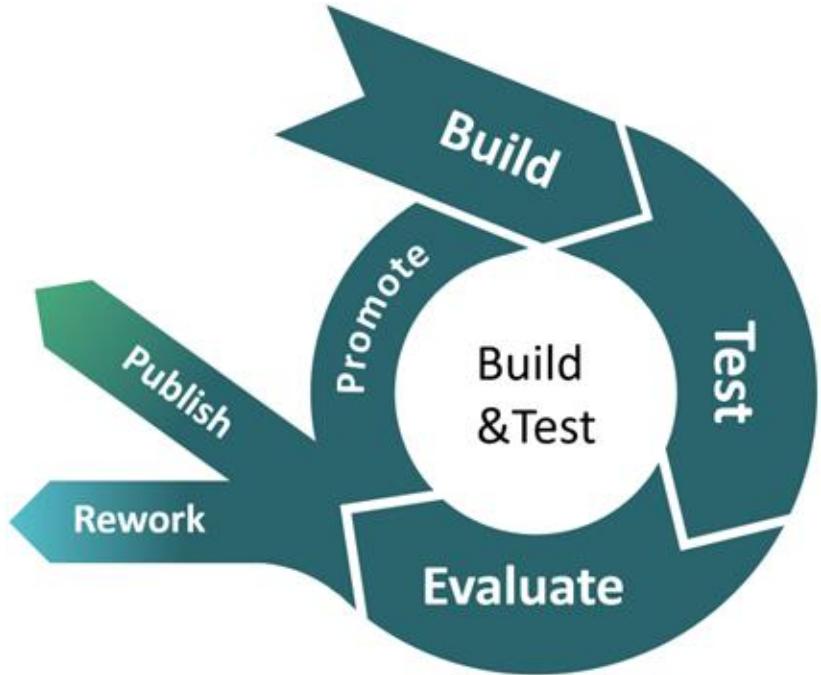


# Server Generation



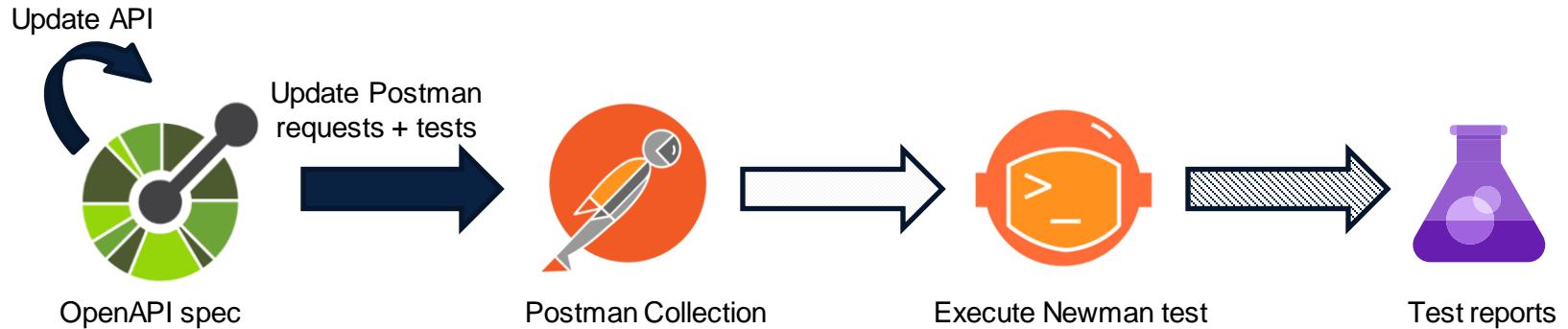
<https://openapi-generator.tech>





• • • • •

# Contract testing



# Portman



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

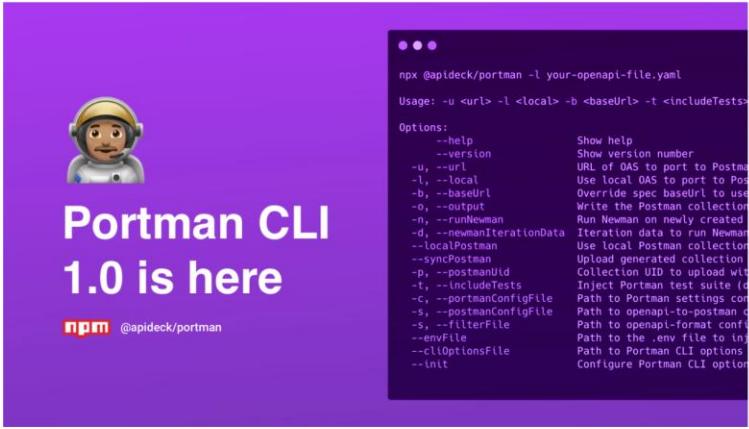
**npm** Search packages Search Sign Up Sign In

---

**@apideck/portman** ts

1.26.2 • Public • Published 3 days ago

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



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 collection
--newmanIterationData	Iteration data to run Newman with
--localPostman	Upload generated collection to local Postman 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
-c, --cliOptionsFile	Path to Portman CLI options config file
--init	Configure Portman CLI options

Install `npm i @apideck/portman`

---

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

---

Homepage [apideck.com](https://apideck.com)

---

Weekly Downloads  9,097

---

Version	License
1.26.2	Apache-2.0

---

Unpacked Size	Total Files
848 kB	301

---

Portman 

# Newman



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

**npm**

Search packages

Search

Sign Up

---

**newman** DT

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

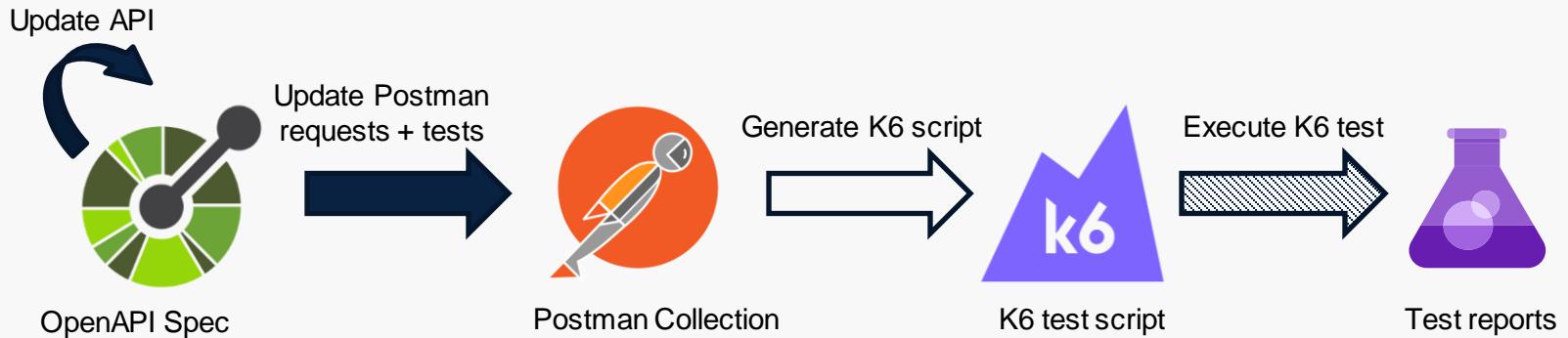
580,509



Version

License

# Performance testing



# K6



<https://k6.io/>

[OPEN SOURCE](#)[GRAFANA CLOUD K6](#)[PRICING](#)[DOCUMENTATION](#)[ABOUT ▾](#)[SIGN IN ▾](#)[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 with a dark blue header. The header features the Gatling logo (an orange stylized 'G' icon followed by the word 'Gatling' in white) on the left, and a navigation bar with links: 'Why Gatling? ▾', 'Learn more ▾', 'About ▾', 'Pricing', 'Book a demo', 'Login', and a prominent orange 'Free Trial' button on the right. The background of the page has a subtle network graph pattern. The main content area contains the tagline 'Test. Succeed. Iterate.' in large white text, followed by a description: 'Gatling is a powerful load-testing solution for applications, APIs, and microservices.' Below this is an orange 'Let's try it!' button.

Test. Succeed. Iterate.

Gatling is a powerful load-testing solution for applications, APIs, and microservices.

Let's try it!



Trusted by 100,000+ businesses worldwide

# Jmeter



<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™



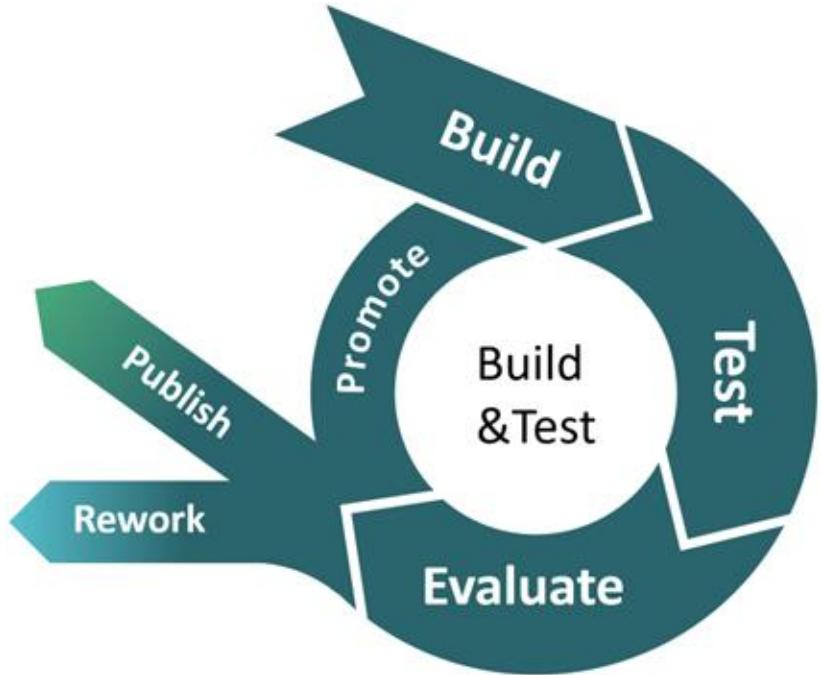
The **Apache JMeter™** application is open source software, a 100% pure Java application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

### 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 ⏺ ×

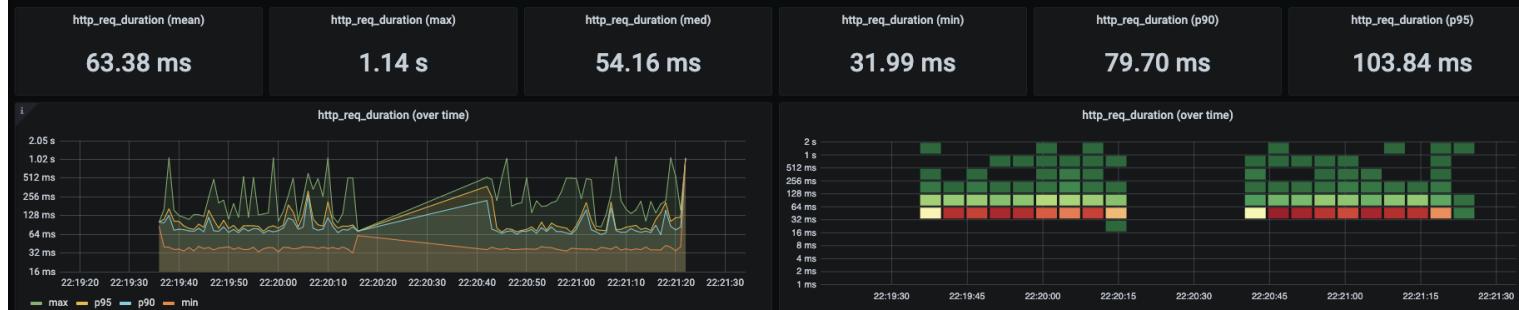
Test	Duration	Failing since	Failing build	Tags
✓ ✅ Newman test suite "List" - buzz-main env (14/14)	0:00:03.194			
✓ [POST] /lists/v2/audiences/{audienceId}/import - Response header Operation-Location is present	0:00:03.196			
✓ [POST] /lists/v2/audiences/{audienceId}/import - Status code is 2xx	0:00:03.194			
✓ [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

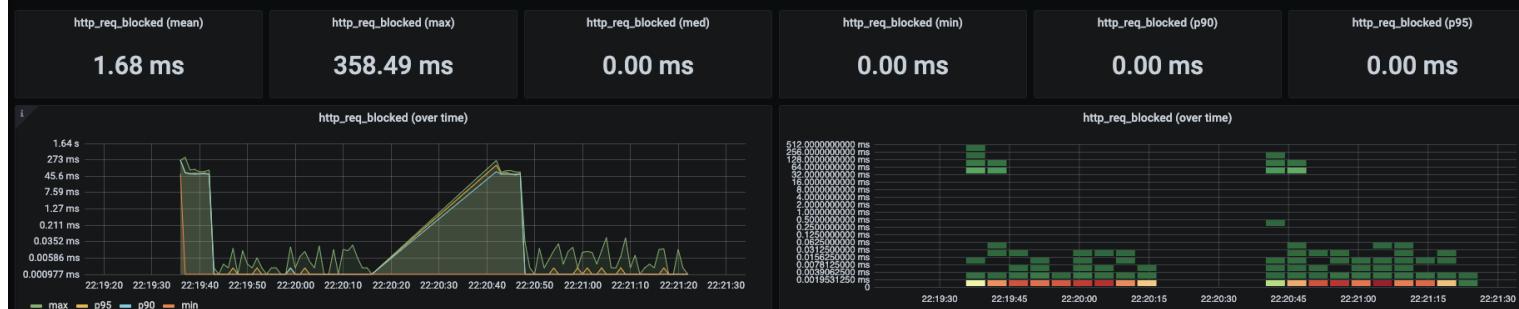
## Dashboard Row



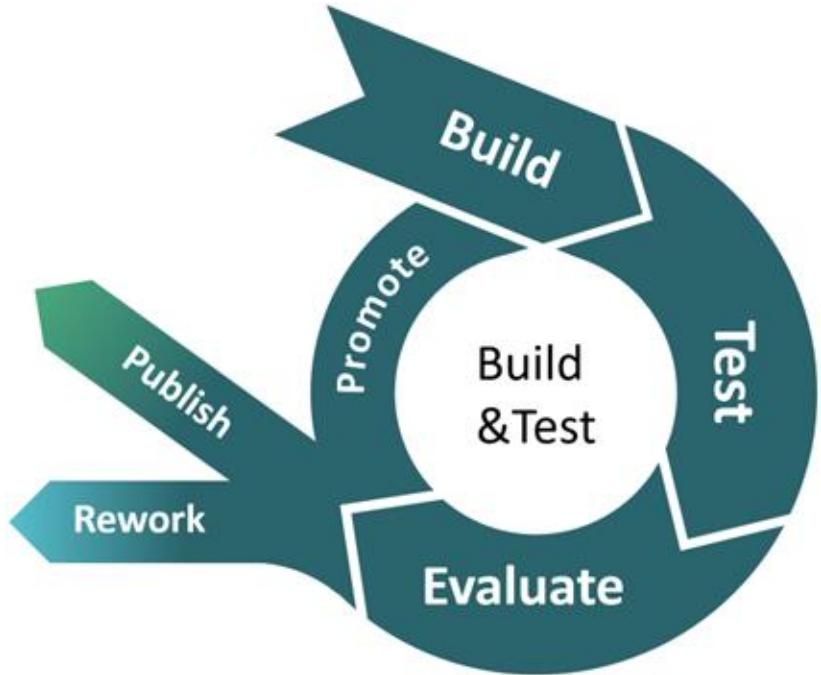
## http\_req\_duration



## http\_req\_blocked

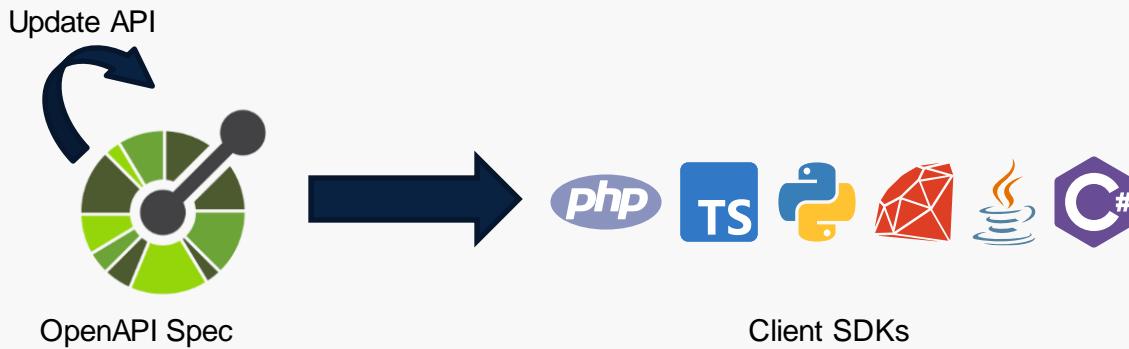


## Dashboard Row



• • • • •

# Generate SDK

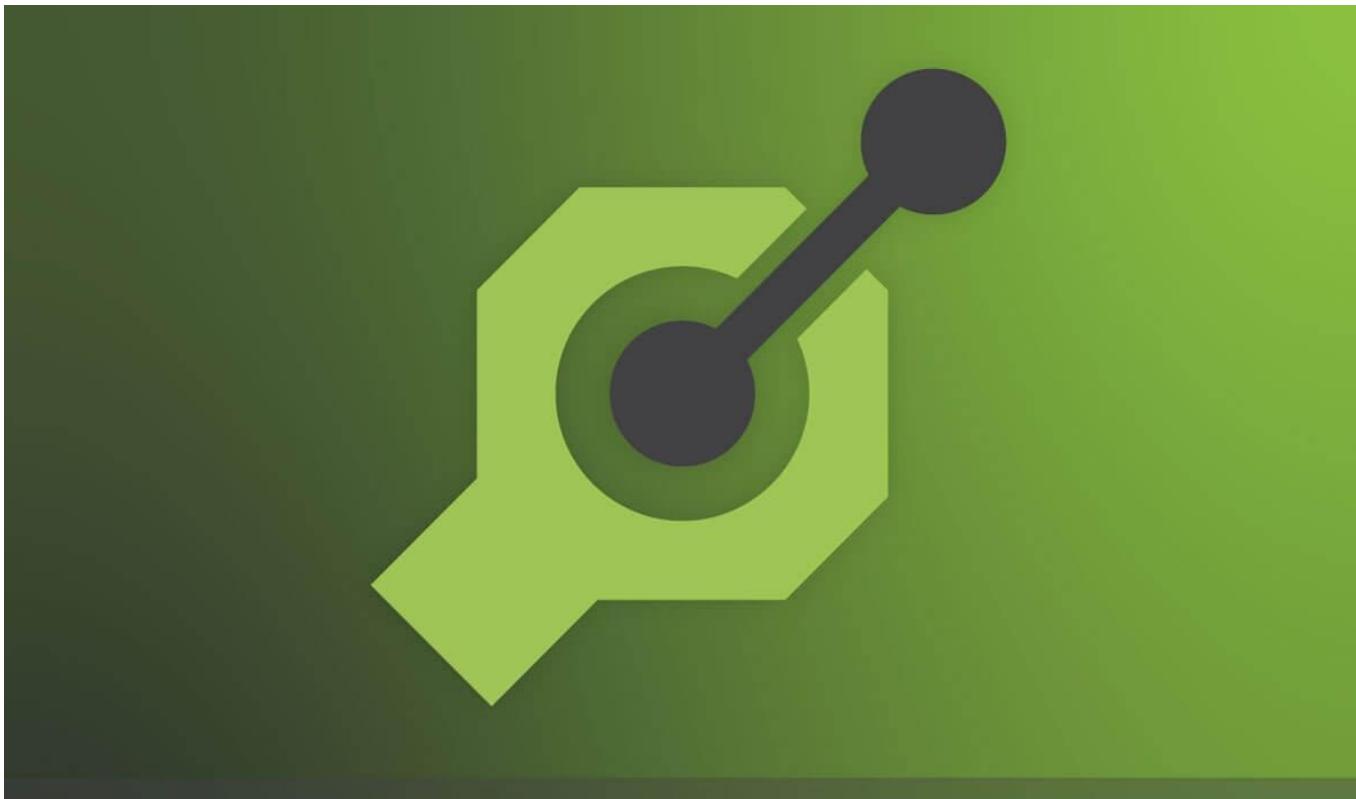


... : : : :

# SDK Client Generation



<https://openapi-generator.tech>



# SDK Client Generation



<https://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 links: "OVERVIEW What is Kiota?", "TRAINING Kiota docs", and "GET STARTED Get Kiota". Each link has a corresponding icon: a person icon for Overview, a computer monitor icon for Training, and a battery icon for Getting Started.

**Microsoft OpenAPI documentation**

Learn how to work with OpenAPI-described APIs using Microsoft tools.

**OVERVIEW**  
What is Kiota?

**TRAINING**  
Kiota docs

**GET STARTED**  
Get Kiota

## Kiota

Kiota builds API clients from OpenAPI descriptions

The screenshot displays a grid of eight cards, each representing a different programming language or framework that Kiota supports. Each card includes the language/framework logo, the name, and two links: "Quick start" and "Azure auth tutorial".

<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 API build tool interface. At the top, there's a navigation bar with links for Products, Showcase, Documentation, Pricing, and Blog, along with a "Get started" button. The main heading is "Input your OpenAPI. Output SDKs and Docs." Below it, the sub-headline reads "Beautiful SDKs and Docs for your API". There are two calls-to-action: "Book a demo" and "Get started - free →". A large inset window displays the Vellum API Reference for the "List Documents" endpoint. It includes a search bar, a sidebar with navigation links like Help Center, Changelog, API Reference, and Introduction, and a detailed view of the API endpoint parameters and a sample code snippet for Node.js. To the right of the inset, there's a sidebar titled "Trusted by" featuring logos for 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

APF 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 (HTTP)

```
import { VellumClient } from "vellum-ai"
const vellum = VellumClient.api.key="API_KEY")
result = await vellum.documents.list({
  documentIndexId: "629e6235",
  limit: 10,
  offset: 0,
  ordering:"label"
})
```

RESPONSE

```
{
  "count": 10,
  "next": "https://api.vellum.ai/next?page=2",
  "previous": null,
  "results": [
    {
      "id": "629e6235-88c6-25e-97be-cdb419fddd52",
      "external_id": "doc_e625e15c",
      "last_uploaded_at": "2023-09-27T08:03:22Z",
      "label": "Test Document"
    }
  ]
}
```

Trusted by

Flatfile

MERGE

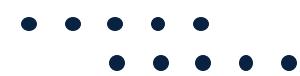
vellum

AssemblyAI

candidhealth

Primer

# SDK Client Generation



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

The screenshot shows the Speakeeasy API client generation interface. 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 main heading reads 'Your API: Enterprise ready in a click'. A subtext below it 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: 'TRY IT NOW →' and 'BOOK A DEMO'. The background features a dark grid pattern with two large, glowing 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.

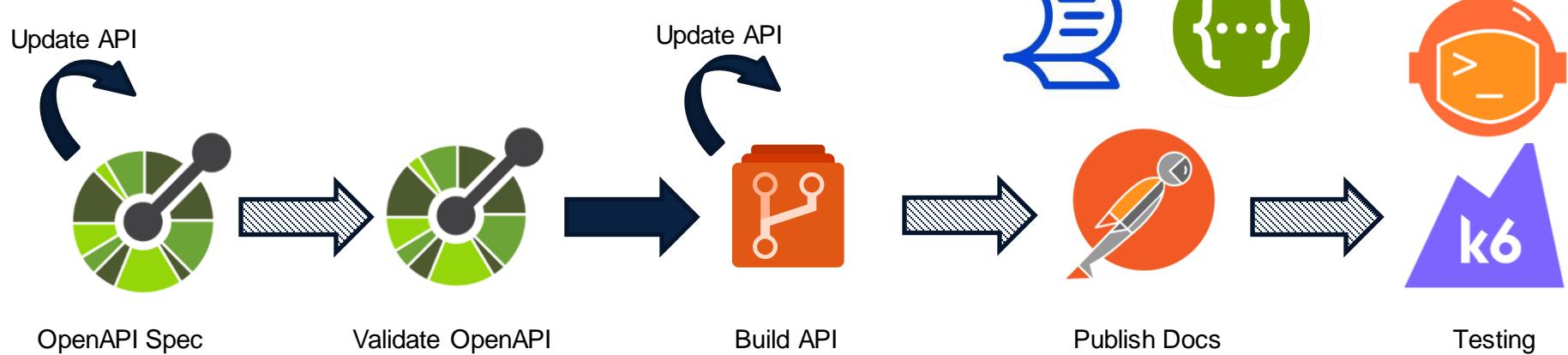
TRY IT NOW → BOOK A DEMO

TRUSTED BY



• • • • •

# Pipeline



...  
• • • • •



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

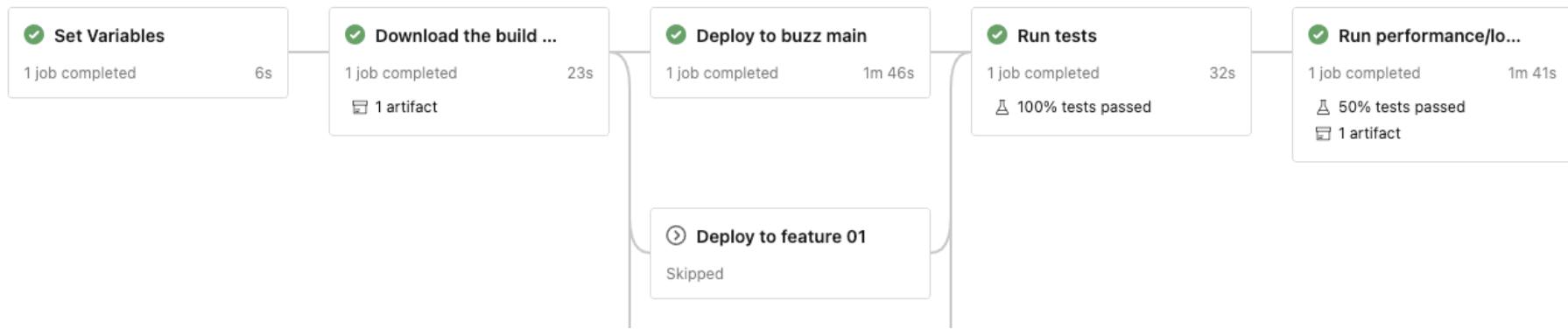


HOOKS

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



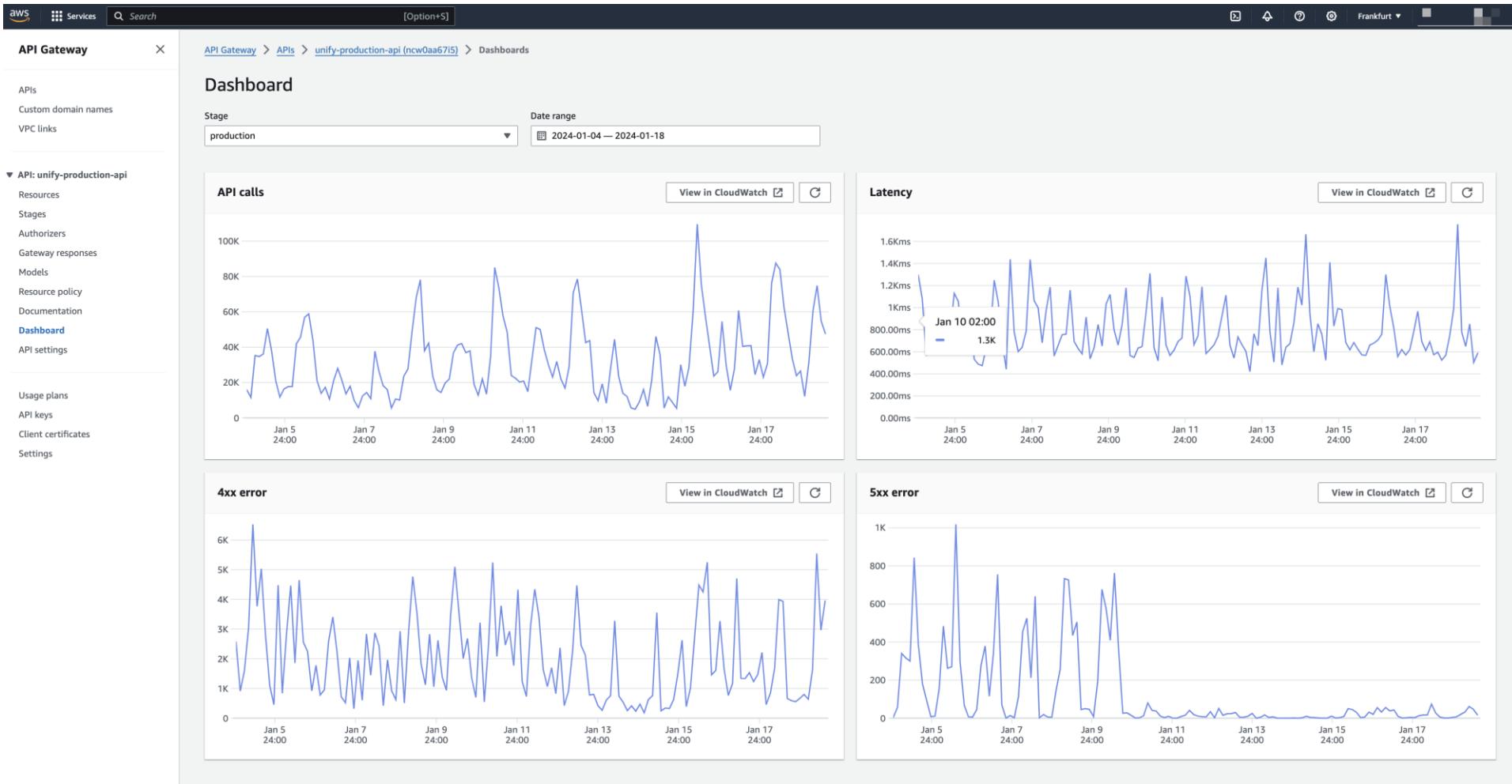
## ✓ Newman :: Filter lists endpoints for Newman test suite



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
```



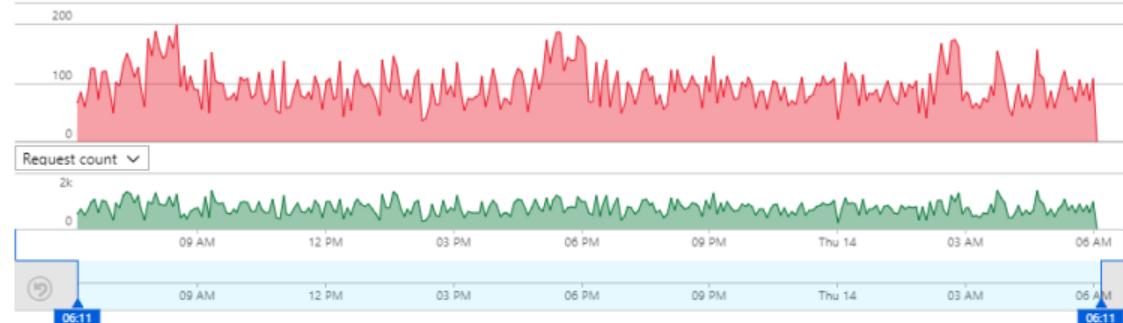
Last 24 hours Roles = All

Server Browser

Operations Dependencies Exceptions Roles

View in Analytics Feedback

Failed request count



Select operation

Search to filter items...

OPERATION NAME	USERS	COUNT (FAILED)	COUNT	
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...

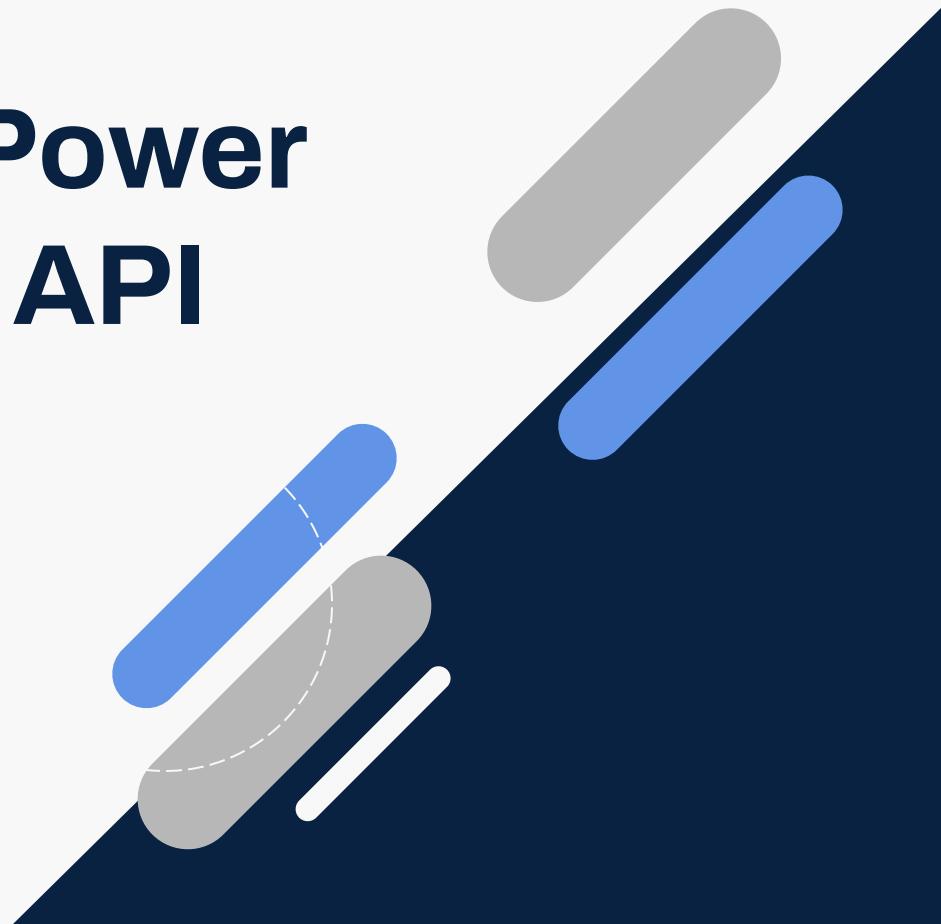
27.53k Operations

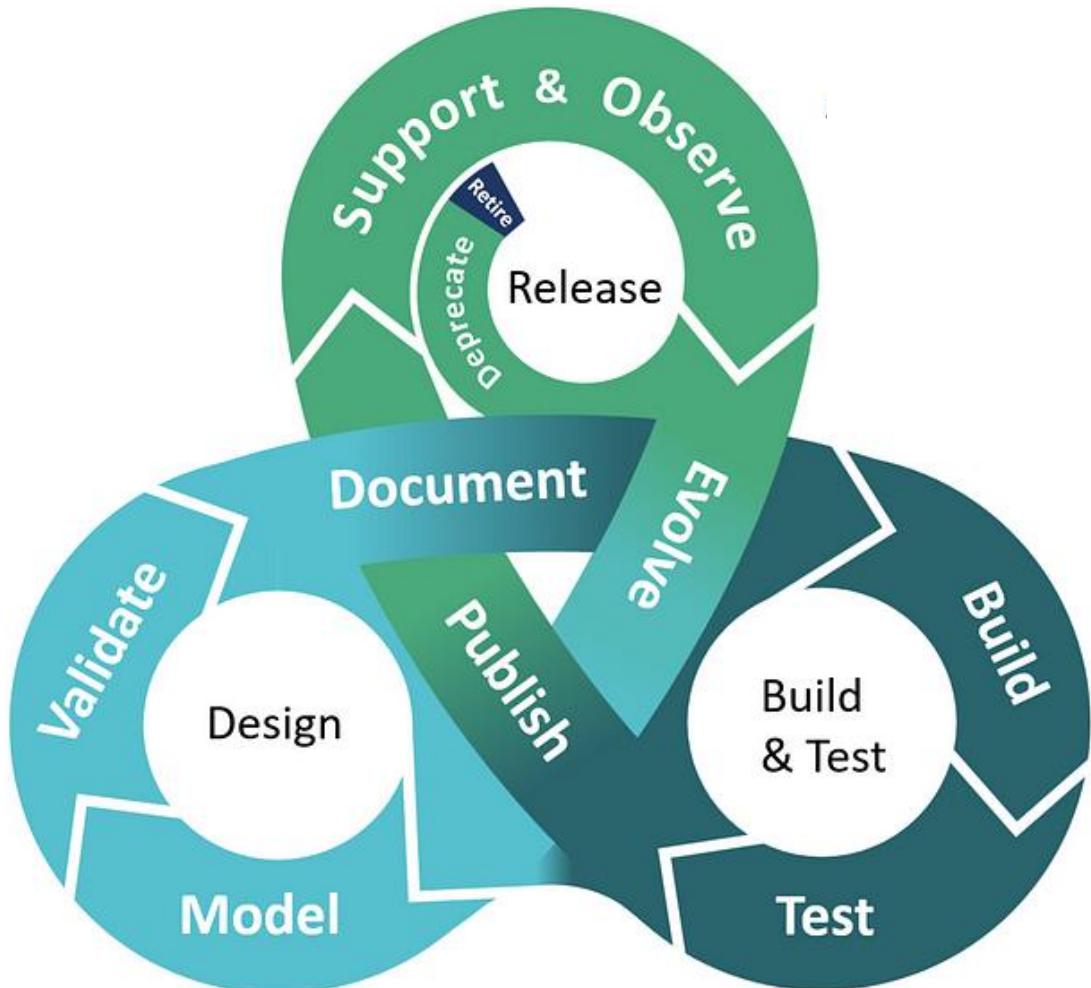


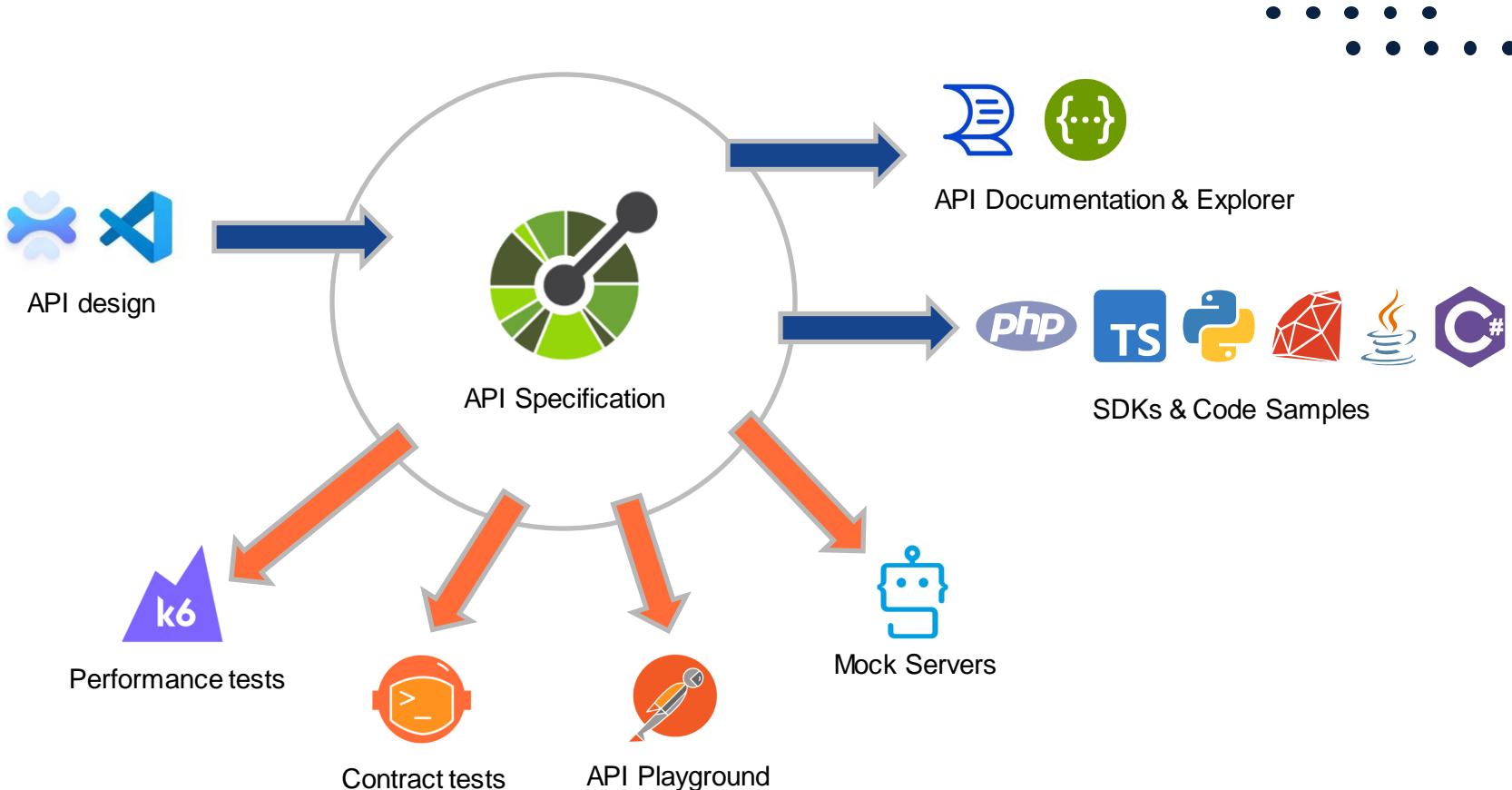
# Unlocking the Power of Spec-Driven API Development

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?

