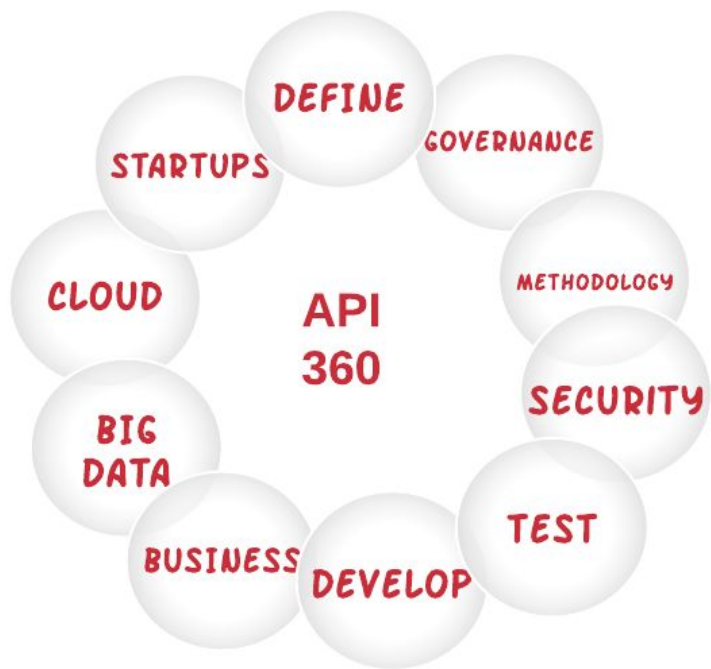


{api⚙️addicts}



3K
9K

2K

+35
meetups

+30k
visualizations

+1,5K
api addicts

Sobre “me”



apiaddicts/users/me

```
{
  "name": "Omar del Valle Rodriguez",
  "email": "omar@odelvalle.com",
  "blog": "blog.odelvalle.com",
  "socialnetwork":
    {
      "twitter": "omarvr72",
      "linkedin": "omardelvalle"
    }
}
```

/cloudappi

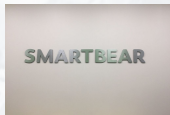
{api⚙️addicts}

1. ¿Qué es Open API Specification?
2. Lenguajes de definición de APIs
3. Esquema y descripción de APIs en OpenAPI
4. ¿Cómo nos ayuda?
5. ¿Cómo introducir Open API en nuestro roadmap de desarrollo?
6. Una API en 10 minutos
7. Preguntas*
8. (¿Cervezas?)

- 
- 2010: Swagger

w♥rdnik

- 2015: Swagger



- 2015: Open Api Initiative



Microsoft

- 2016: Swagger Specification - Open API Specification

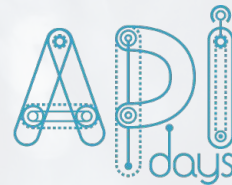


- 2017: Open API Specification v3.0.0

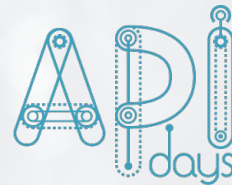


MuleSoft

RAML (RESTful API Modeling language)



Open API Specifications define un estándar que permite tanto a humanos como a ordenadores, conocer y entender las capacidades de un servicio sin necesidad de acceder al código fuente o documentación.



Lenguajes de definición de APIs {api^{⚙️}addicts}

WADL

```
<application xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wadl.dev.java.net/2009/02 wadl.xsd"
  xmlns:tns="urn:yahoo:yn" xmlns:yn="urn:yahoo:yn" xmlns:ya="urn:yahoo:api"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://wadl.dev.java.net/2009/02">
  <grammars>
    <include href="NewsSearchResponse.xsd"/>
    <include href="Error.xsd"/>
  </grammars>

  <resources base="http://api.search.yahoo.com/NewsSearchService/V1/">
    <resource path="newsSearch">
      <method name="GET" id="search">
        <request>
          <param name="appid" type="xsd:string" style="query" required="true"/>
          <param name="query" type="xsd:string" style="query" required="true"/>
          <param name="type" style="query" default="all">
            <option value="all"/>
            <option value="any"/>
            <option value="phrase"/>
          </param>
          <param name="results" style="query" type="xsd:int" default="10"/>
          <param name="start" style="query" type="xsd:int" default="1"/>
          <param name="sort" style="query" default="rank">
            <option value="rank"/>
            <option value="date"/>
          </param>
          <param name="language" style="query" type="xsd:string"/>
        </request>
        <response status="200">
          <representation mediaType="application/xml" element="yn:ResultSet"/>
        </response>
        <response status="400">
```

```
<?xml version="1.0" encoding="UTF-8"?>
<description xmlns="http://www.w3.org/ns/wsdl"
  xmlns:tns="http://www.tmsws.com/wsdl20sample"
  xmlns:whhttp="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:wsoap="http://schemas.xmlsoap.org/wsdl/soap/"
  targetNamespace="http://www.tmsws.com/wsdl20sample">

  <documentation>
    This is a sample WSDL 2.0 document.
  </documentation>

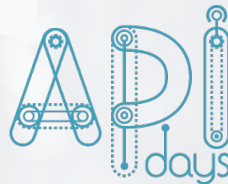
  <!-- Abstract type -->
  <types>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns="http://www.tmsws.com/wsdl20sample"
      targetNamespace="http://www.example.com/wsdl20sample">

      <xs:element name="request"> ... </xs:element>
      <xs:element name="response"> ... </xs:element>
    </xs:schema>
  </types>

  <!-- Abstract interfaces -->
  <interface name="Interface1">
    <fault name="Error1" element="tns:response"/>
    <operation name="Get" pattern="http://www.w3.org/ns/wsdl/in-out">
      <input messageLabel="In" element="tns:request"/>
      <output messageLabel="Out" element="tns:response"/>
    </operation>
  </interface>

  <!-- Concrete Binding Over HTTP -->
  <binding name="HttpBinding" interface="tns:Interface1"
    type="http://www.w3.org/ns/wsdl/http">
    <operation ref="tns:Get" whhttp:method="GET"/>
  </binding>
```

WSDL

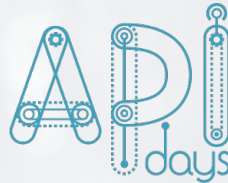


Lenguajes de definición de APIs {api⚙️addicts}

```
1 swagger: '2.0'
2 - info:
3   description: |
4     This is a sample server Petstore server. You can find
5     out more about Swagger at
6     [http://swagger.io](http://swagger.io) or on
7     [irc.freenode.net, #swagger](http://swagger.io/irc/).
8   version: 1.0.0
9   title: Swagger Petstore
10  termsOfService: http://swagger.io/terms/
11  contact:
12    email: apiteam@swagger.io
13  license:
14    name: Apache 2.0
15    url: http://www.apache.org/licenses/LICENSE-2.0.html
16  # host: petstore.swagger.io
17  # basePath: /v2
18  tags:
19  - name: pet
20    description: Everything about your Pets
21    externalDocs:
22      description: Find out more
23      url: http://swagger.io
24  - name: store
25    description: Access to Petstore orders
26  - name: user
27    description: Operations about user
28    externalDocs:
29      description: Find out more about our store
30      url: http://swagger.io
31  # schemes:
32  # - http
33  paths:
34  - /pet:
35    post:
36      tags:
37      - pet
38      summary: Add a new pet to the store
39      operationId: addPet
40      consumes:
41      - application/json
42      - application/xml
43      produces:
44      - application/json
```

Schema:

- swagger
- info
- host
- security
- securityDefinitions
- consumes / produces
- paths
- parameters
- responses
- definitions
- ...



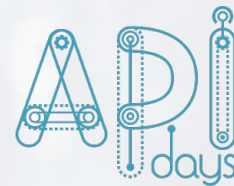
Ejemplos (swagger templates)

- Simple API
- Petstore
- OAuth 2.0

2.0: <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

3.0: <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.0.md>

Pre-release: <https://nordicapis.com/whats-new-in-openapi-3-0/>



Herramientas de trabajo

- Swagger Editor
- Visual Studio Code + Plugging
- Generación de código

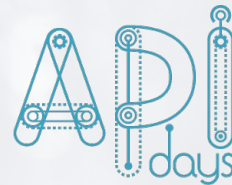
Mocks

- Swagger Hub
- Swagger Server

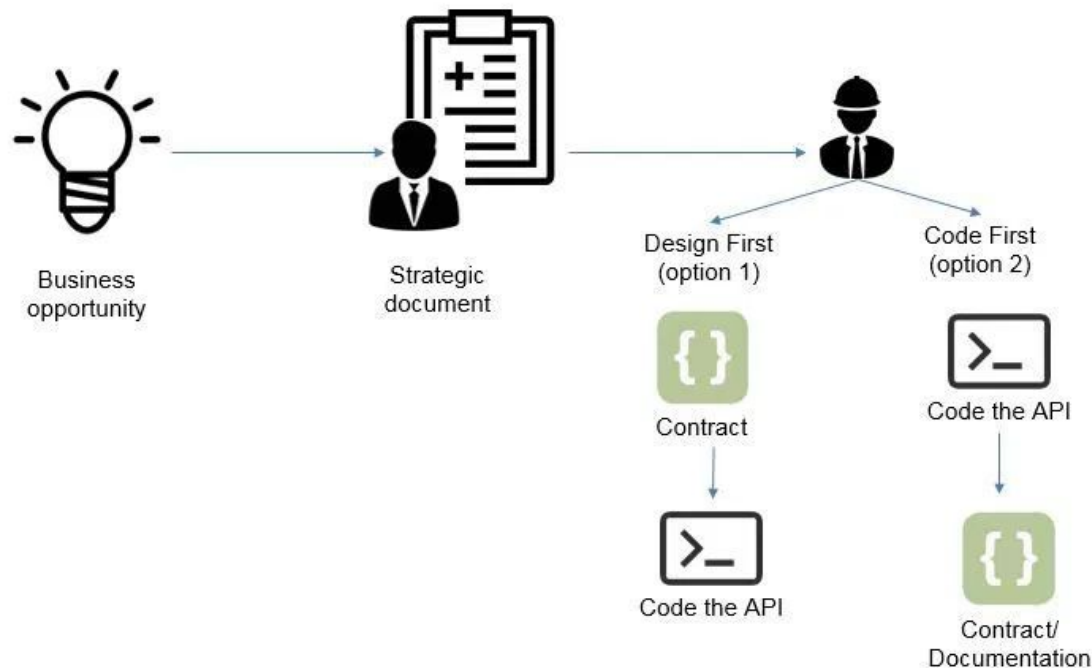
<https://www.npmjs.com/package/swagger-server>

Integraciones

- Apigee
- Amazon Gateway
- Azure API Management
- IBM Connect
- WSO2 (Example)



Design First vs Code First



Design First

- Exposición a terceros
- Microservicios
- Reutilización de recursos
- Desarrollo por contrato

Code First

- APIs internas
- Pocos servicios
- Único equipo de desarrollo
- Inicio inmediato

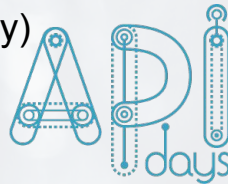
:1 Nuevas API: Estándar de definición de APIs

1. Definición de recursos
2. Política de versionado
3. Definición de Métodos
4. Convención de nombres
5. Seguridad
6. Http code (Response)
7. Respuestas (vacías, paginado, etc.)
8. Errores

<https://cloud.google.com/apis/design/>

:2 APIs existentes (API Managements and goto :1)

1. API Management (Apigee, Azure API Management, WSO2, IBM Connect, Amazon Gateway)
2. <https://pages.apigee.com/rs/apigee/images/api-facade-pattern-ebook-2012-06.pdf>



¿Preguntas?



Beers & Networking

Email:

contacta@apiaddicts.org

Web:

www.apiaddicts.org

Github:

<https://github.com/apiaddicts>

Follow us:**Linkedin:**

<https://www.linkedin.com/company/apiaddicts>

Twitter:

<https://twitter.com/apiaddicts>

Facebook:

<https://www.facebook.com/pages/APIAddicts>

Meetup:

<http://www.meetup.com/es-ES/ApiAddicts>

<http://www.meetup.com/es-ES/ApiAddictsBCN>