



# Consumiendo APIs con React y Swagger

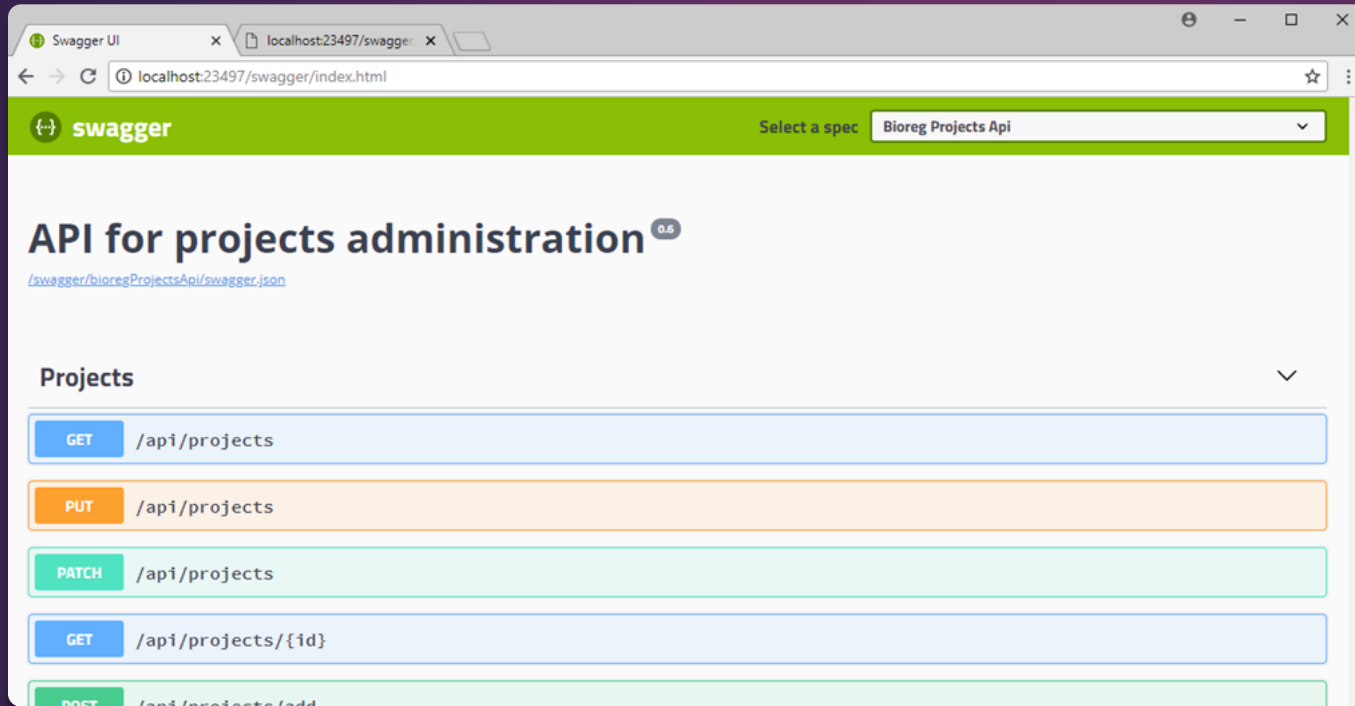
LAUTARO CARRO

Los STRING no se  
compilan

---

# Los STRING no se compilan

- ❑ Se construyen URLs concatenando strings
- ❑ Bugs de 400, 404 y 415
- ❑ ¿Cómo se da cuenta el Frontend de que un endpoint cambió?
- ❑ Cambiar las request de **v1** a **v2**
- ❑ Demasiado error humano



# Swagger y OpenApi

# Swagger

**Swagger** es un conjunto de herramientas de software de código abierto para diseñar, construir, documentar y utilizar servicios REST.

- ▶ Documentación Automatizada
- ▶ Generación de Código
- ▶ Generación de Casos de Prueba

# OpenApi

**OpenAPI** es un estándar para la descripción de WebApis.

La especificación **OpenAPI** define un formato de descripción abierto e independiente de los fabricantes para los servicios de API.

```
{
  "swagger": "2.0",
  "info": {
    "description": "This is a sample server Petstore server. You can",
    "version": "1.0.6",
    "title": "Swagger Petstore",
    "termsOfService": "http://swagger.io/terms/",
    "contact": {
      "email": "apiteam@swagger.io"
    },
    "license": {
      "name": "Apache 2.0",
      "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
    }
  },
  "host": "petstore.swagger.io",
  "basePath": "/v2",
  "tags": [
    {
      "name": "pet",
      "description": "Everything about your Pets",
      "externalDocs": {
        "description": "Find out more",
        "url": "http://swagger.io"
      }
    }
  ],
}
```

# OpenApi

```
{
  "swagger": "2.0",
  "info": {
    "description": "This is a sample server Petstore server. You can",
    "version": "1.0.6",
    "title": "Swagger Petstore",
    "termsOfService": "http://swagger.io/terms/",
    "contact": {
      "email": "apiteam@swagger.io"
    },
    "license": {
      "name": "Apache 2.0",
      "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
    }
  },
  "host": "petstore.swagger.io",
  "basePath": "/v2",
  "tags": [
    {
      "name": "pet",
      "description": "Everything about your Pets",
      "externalDocs": {
        "description": "Find out more",
        "url": "http://swagger.io"
      }
    }
  ],
}
```

- **Info:** versión, nombre, etc. de la API.
- **Contact:** datos de contacto del proveedor de la API.
- **License:** licencia bajo la cual la API proporciona sus datos.

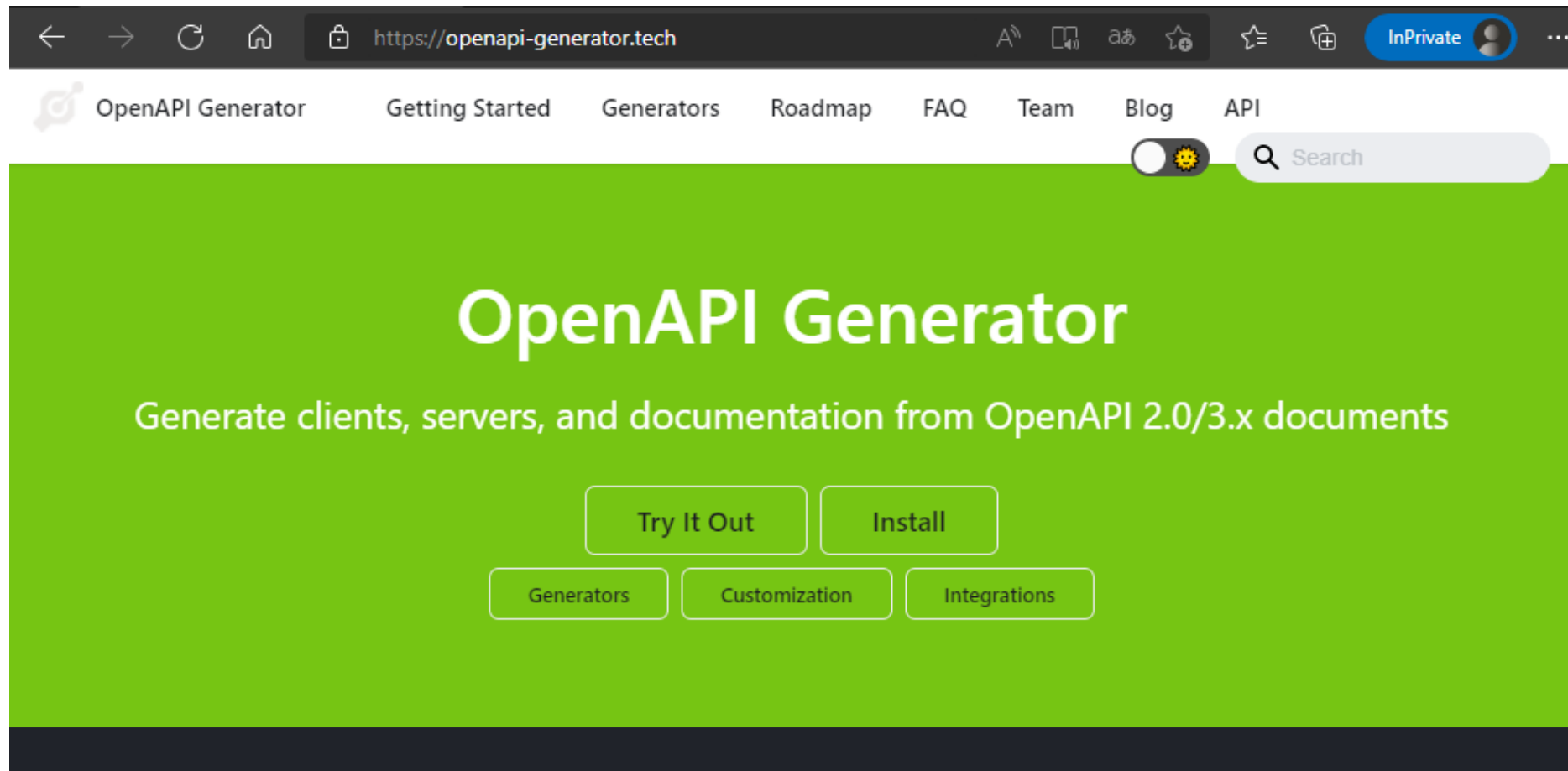
# OpenApi

```
{
  "swagger": "2.0",
  "info": {
    "description": "This is a sample server Petstore server. You can",
    "version": "1.0.6",
    "title": "Swagger Petstore",
    "termsOfService": "http://swagger.io/terms/",
    "contact": {
      "email": "apiteam@swagger.io"
    },
    "license": {
      "name": "Apache 2.0",
      "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
    }
  },
  "host": "petstore.swagger.io",
  "basePath": "/v2",
  "tags": [
    {
      "name": "pet",
      "description": "Everything about your Pets",
      "externalDocs": {
        "description": "Find out more",
        "url": "http://swagger.io"
      }
    }
  ],
}
```

- ▶ **Paths:** rutas relativas a los puntos finales de la API que se utilizan junto con el servidor del objeto.
- ▶ **Path Item:** operaciones permitidas para una ruta específica como GET, PUT, POST, DELETE.
- ▶ **Components:** componentes encapsulados que pueden utilizarse varias veces dentro de una definición de API.
- ▶ **Operation:** especifica, entre otras cosas, los parámetros y las respuestas del servidor que se esperan de una operación.



# Generación de Código



# Generación de Código



**OPENAPI**  
INITIATIVE

```
# install the latest version of "openapi-generator-cli"  
npm install @openapitools/openapi-generator-cli -g  
  
# use a specific version of "openapi-generator-cli"  
openapi-generator-cli version-manager set 5.3.0  
  
# Or install it as dev-dependency in your node.js projects  
npm install @openapitools/openapi-generator-cli -D
```



# Generación de Código

```
...[HttpGet(Name = "GetUsers")]
...[HttpGet(template: "{id}", Name = "GetUserById")]
...[HttpPost(Name = "CreateUser")]
...[HttpPut(Name = "UpdateUser")]
...[HttpDelete(template: "{id}", Name = "DeleteUser")]
```

```
export class UsersApi extends BaseAPI {
  ... /** ...
  ... public createUser(createUserInput?: CreateUserInput,
  ... }
  ... /** ...
  ... public deleteUser(id: number, options?: AxiosRequest
  ... }
  ... /** ...
  ... public getUserById(id: number, options?: AxiosReques
  ... }
  ... /** ...
  ... public getUsers(options?: AxiosRequestConfig) { ...
  ... }
  ... /** ...
  ... public updateUser(userDto?: UserDto, options?: Axios
  ... }
}
```

LIVE DEMO



# Beneficios

- ❑ Evitamos Bugs
- ❑ Mejoramos la calidad de código
- ❑ Mejoramos la experiencia de desarrollo
- ❑ Mantenemos la flexibilidad de Axios y se puede utilizar con otros patrones (Ej: Redux)
- ❑ El proyecto de Backend y Frontend siempre sincronizados



# Recomendaciones

- ❑ Ejecutar `openapi-generator-cli` en CI/CD
- ❑ Antes de generar código, asegurarnos de que SwaggerUI funcione
- ❑ Agregar al “gitignore” el archivo `openapi.json`
- ❑ No tocar el código generado
- ❑ Token JWT: Agregar lógica en el `apiClient.ts`

# Para pensar

¿PORQUE NO NOS ANIMAMOS A USAR HERRAMIENTAS DE AUTOMATIZACIÓN EN PRODUCCIÓN?



@lauchacarro



Lautarocarro.blog

algeiba 

# Muchas Gracias

CONSUMIENDO APIS CON REACT Y  
SWAGGER

