



El futuro digital
es de todos

MinTIC

Misión
TIC 2022



Diseño de una API RESTful

Jefferson A. Peña Torres

jefferson.amado.pena@correounivalle.edu.co



El futuro digital
es de todos

MinTIC



Desarrolla una API RESTful

Agenda



01

¿Que es una API?

Funcionalidades que
ofrece un programa

Ejemplo de una API



02

Arquitectura REST

Sigue esta arquitectura
para crear un
microservicio

Ejemplo de gráfico



03

Implementa REST

Diseño de un servicio
RESTful

Ejemplo de un diseño



Desarrolla una API RESTful

¿Que es un API?

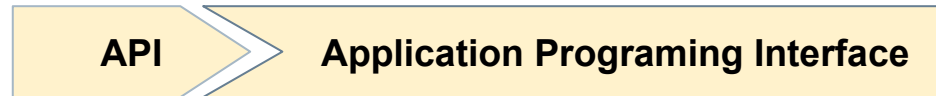
API

Application Programing Interface

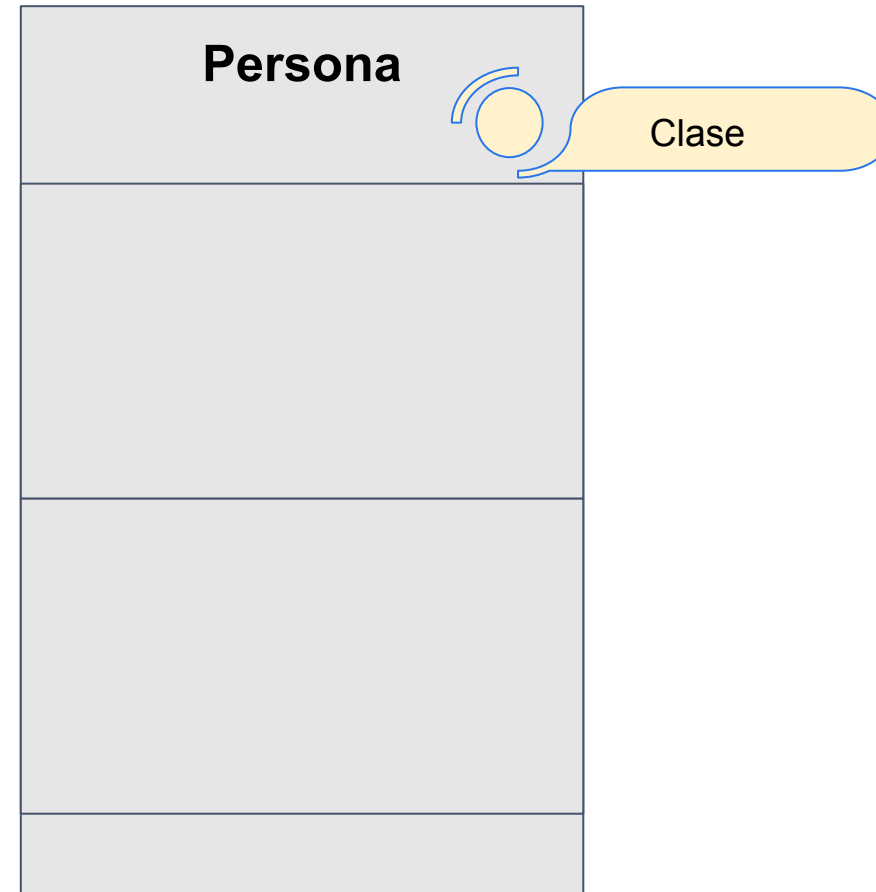
Son las operaciones/funcionalidades que nos ofrece un tercero para interactuar con una herramienta/programa o software.

Desarrolla una API **RESTful**

¿Que es un API?



Son las operaciones/funcionalidades que nos ofrece un tercero para interactuar con una herramienta/programa o software.



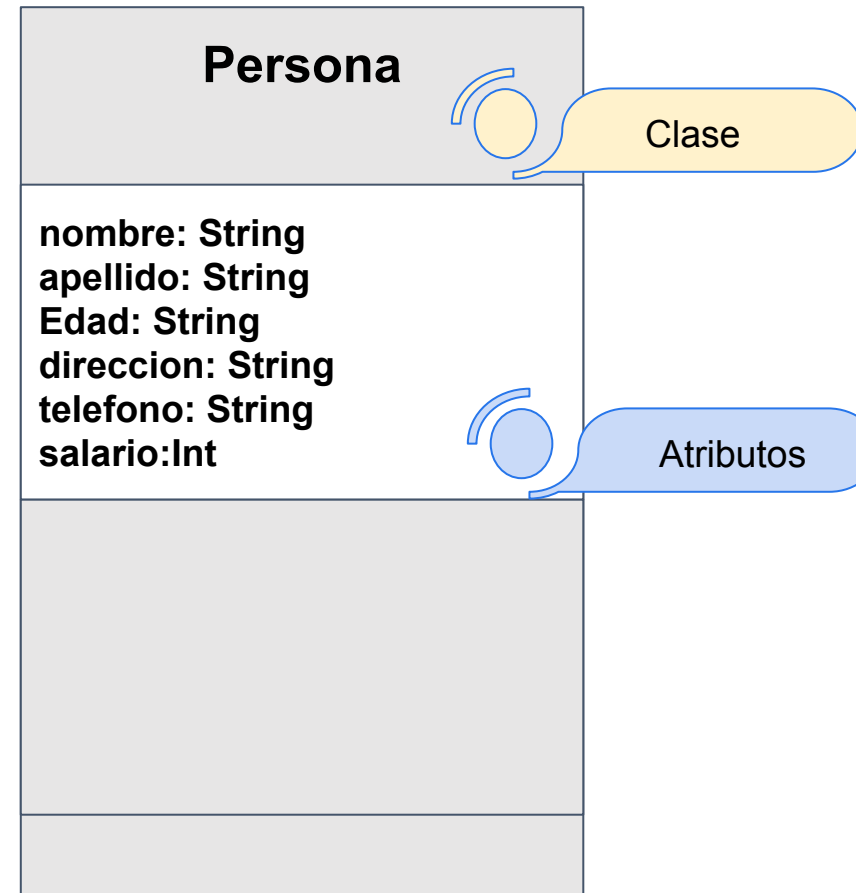
Desarrolla una API RESTful

¿Que es un API?

API

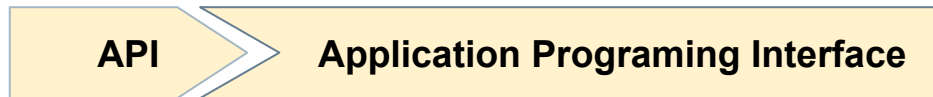
Application Programing Interface

Son las operaciones/funcionalidades que nos ofrece un tercero para interactuar con una herramienta/programa o software.

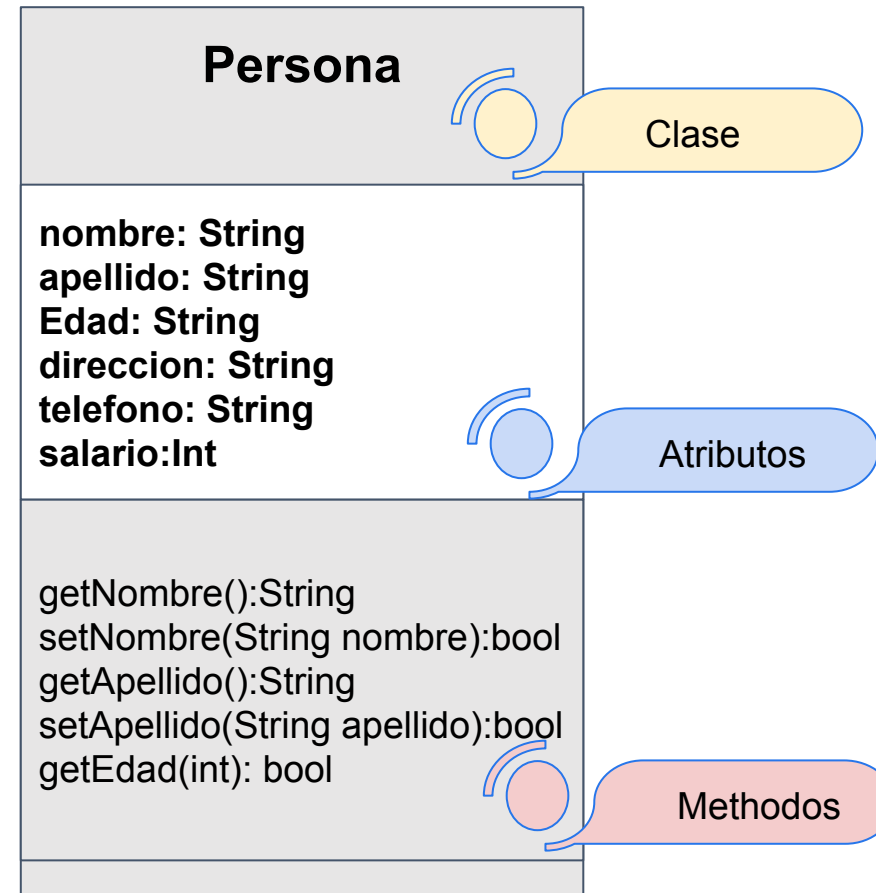


Desarrolla una API **RESTful**

¿Que es un API?

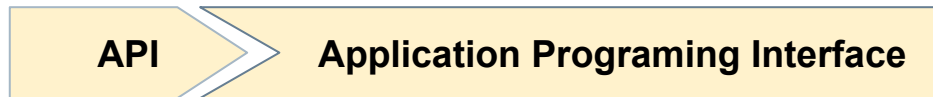


Son las operaciones/funcionalidades que nos ofrece un tercero para interactuar con una herramienta/programa o software.

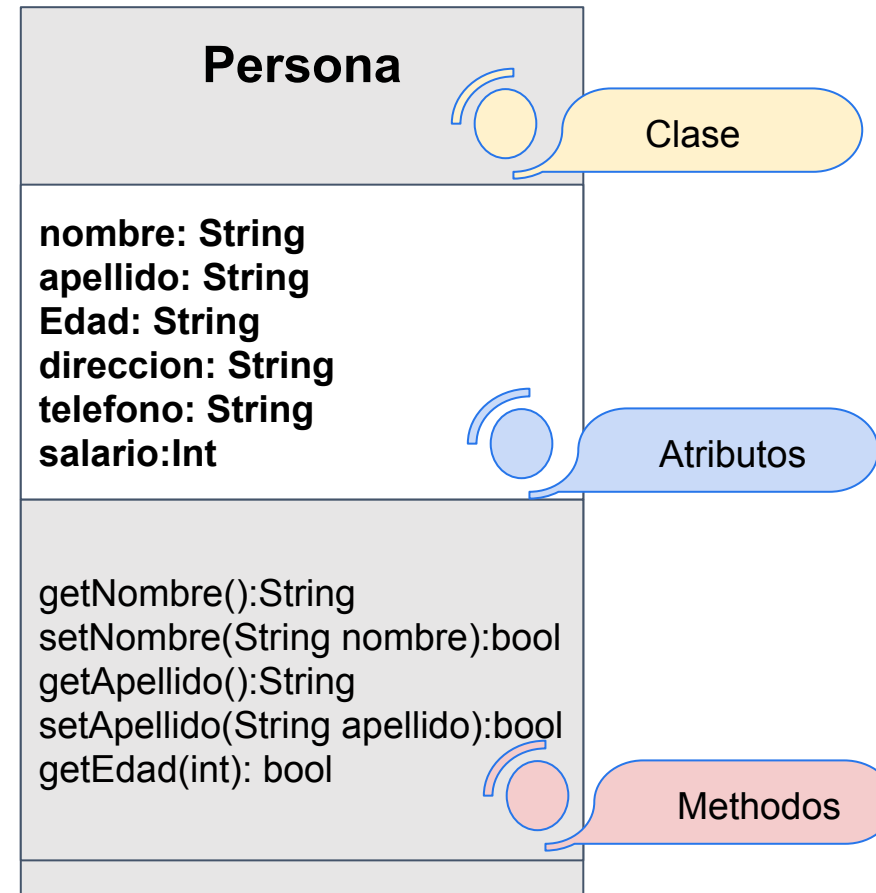


Desarrolla una API **RESTful**

¿Que es un API?



Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.



Desarrolla una API RESTful

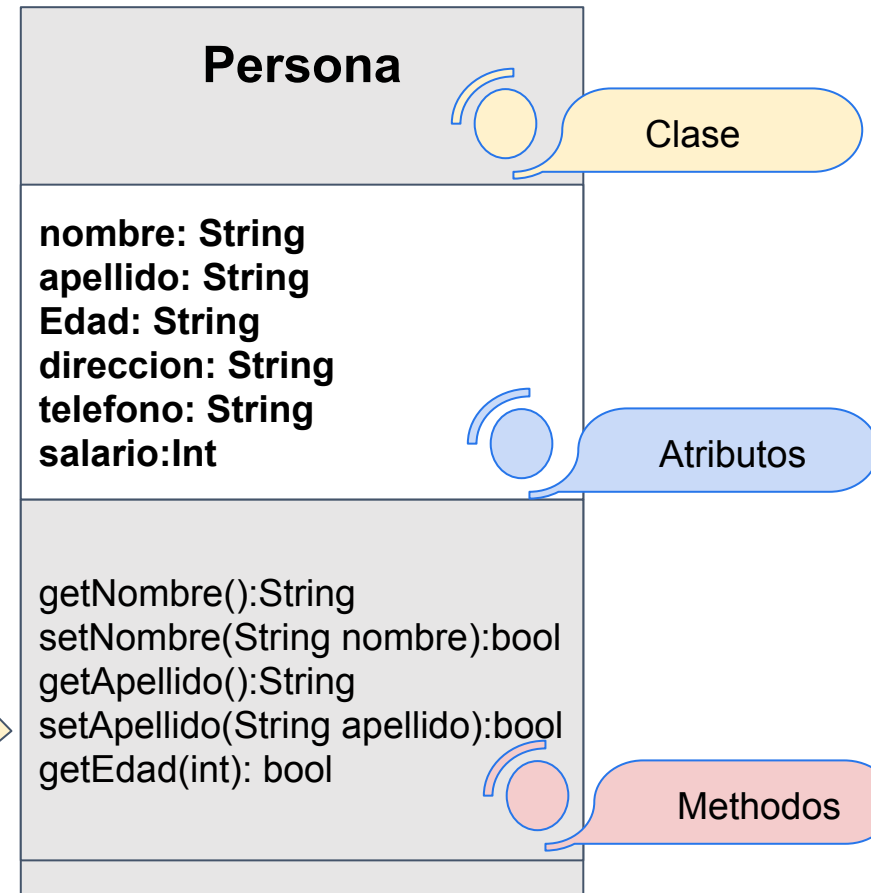
¿Que es un API?

API

Application Programing Interface

Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.

Si un tercero entrega desarrolla la clase persona, estos métodos harán parte de la API



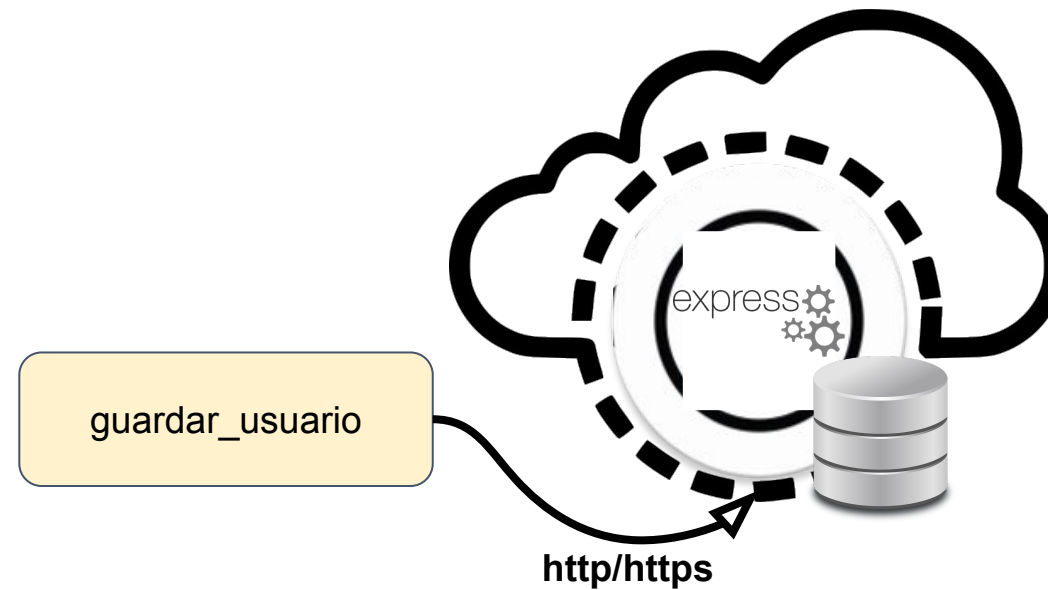
Desarrolla una API RESTful

¿Que es un API?

API

Application Programing Interface

Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.



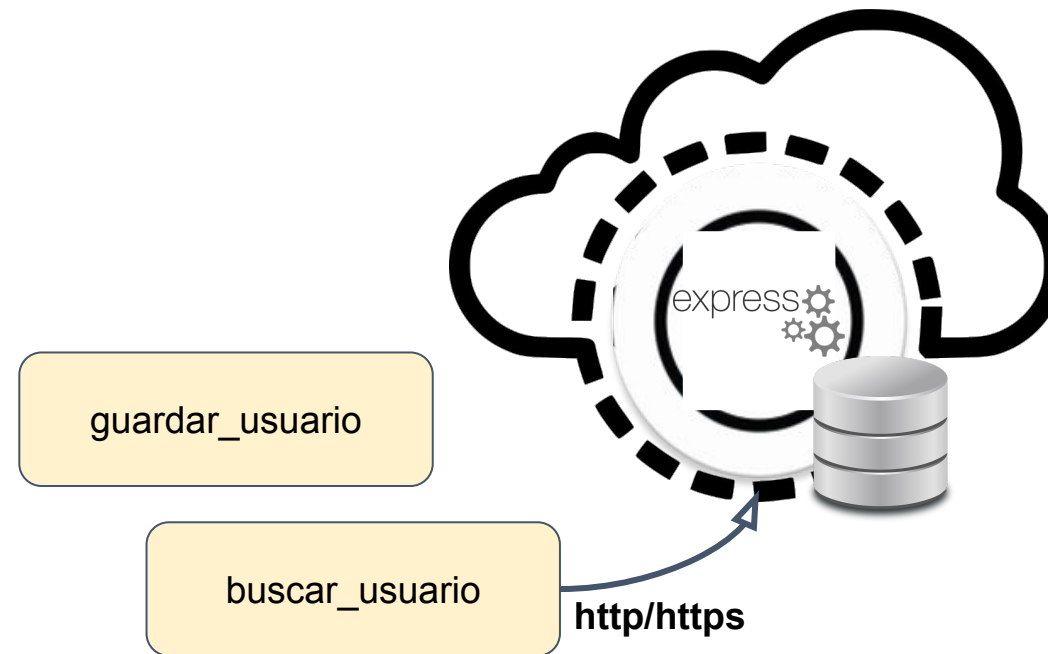
Desarrolla una API RESTful

¿Que es un API?

API

Application Programing Interface

Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.



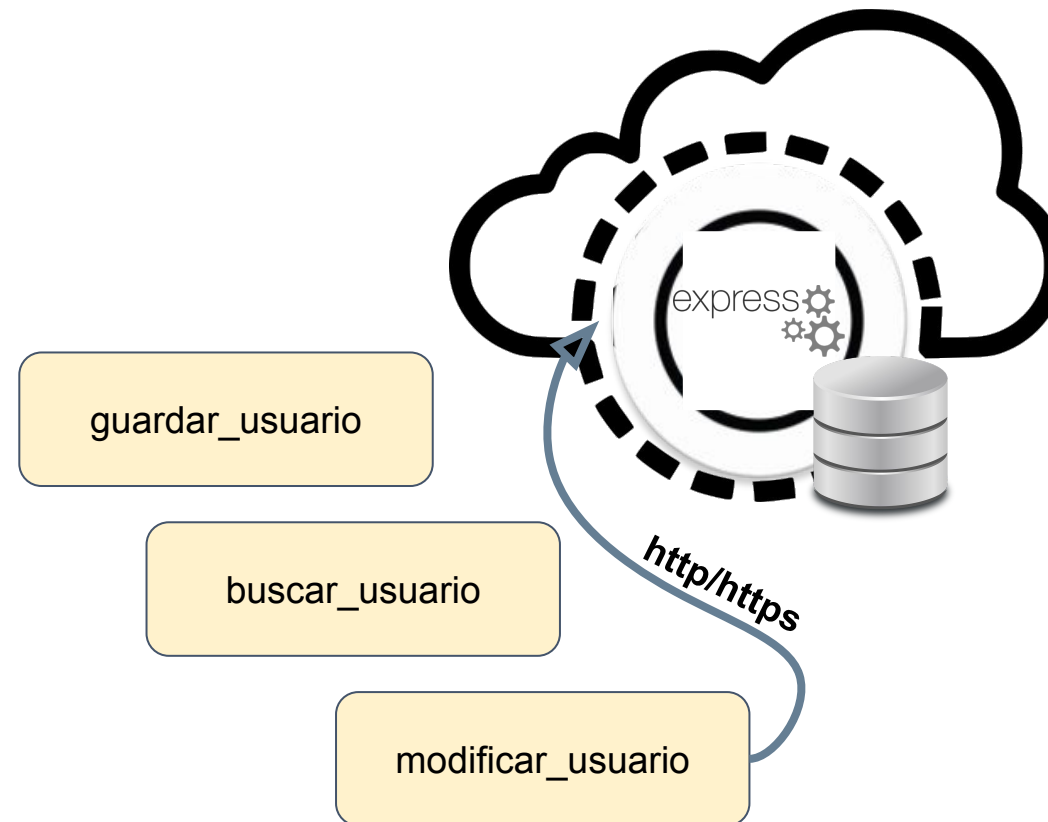
Desarrolla una API RESTful

¿Que es un API?

API

Application Programing Interface

Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.



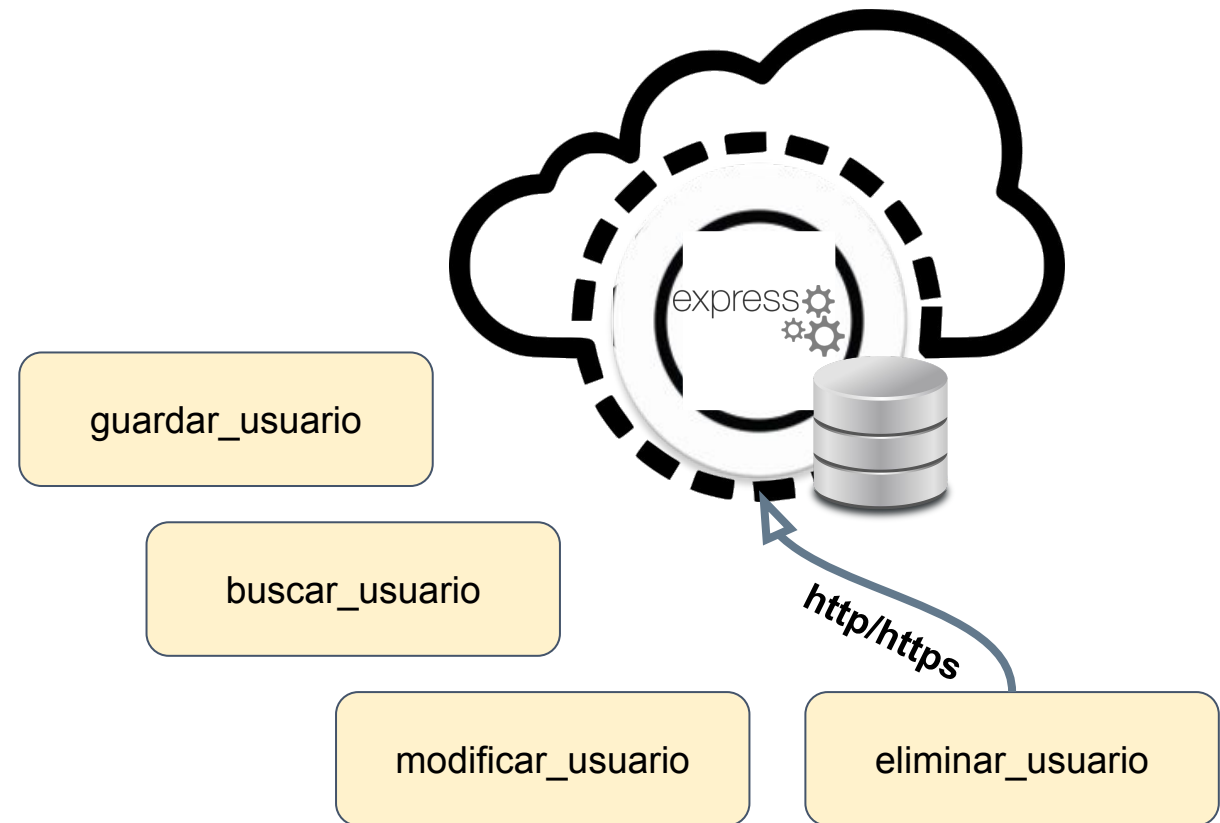
Desarrolla una API RESTful

¿Que es un API?

API

Application Programing Interface

Son las **operaciones/funcionalidades** que nos ofrece un **tercero** para interactuar con una herramienta/programa o software.



Desarrolla una API RESTful

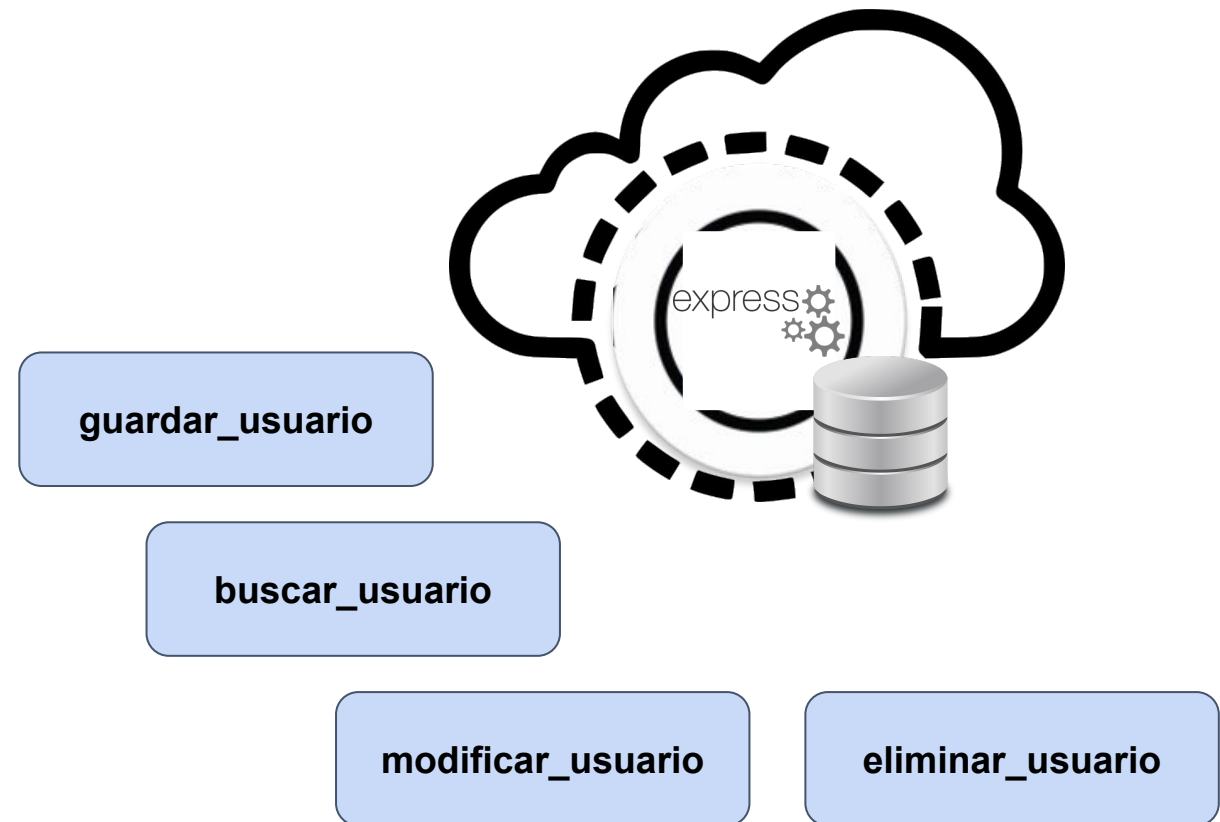
Arquitectura REST

REST

Representational State Transfer

Es **un estilo** de arquitectura de software que se creó para guiar el diseño y desarrollo de la arquitectura para la World Wide Web.

- Escalabilidad
- Interfaces uniformes
- Creación de servicios web



Desarrolla una API RESTful

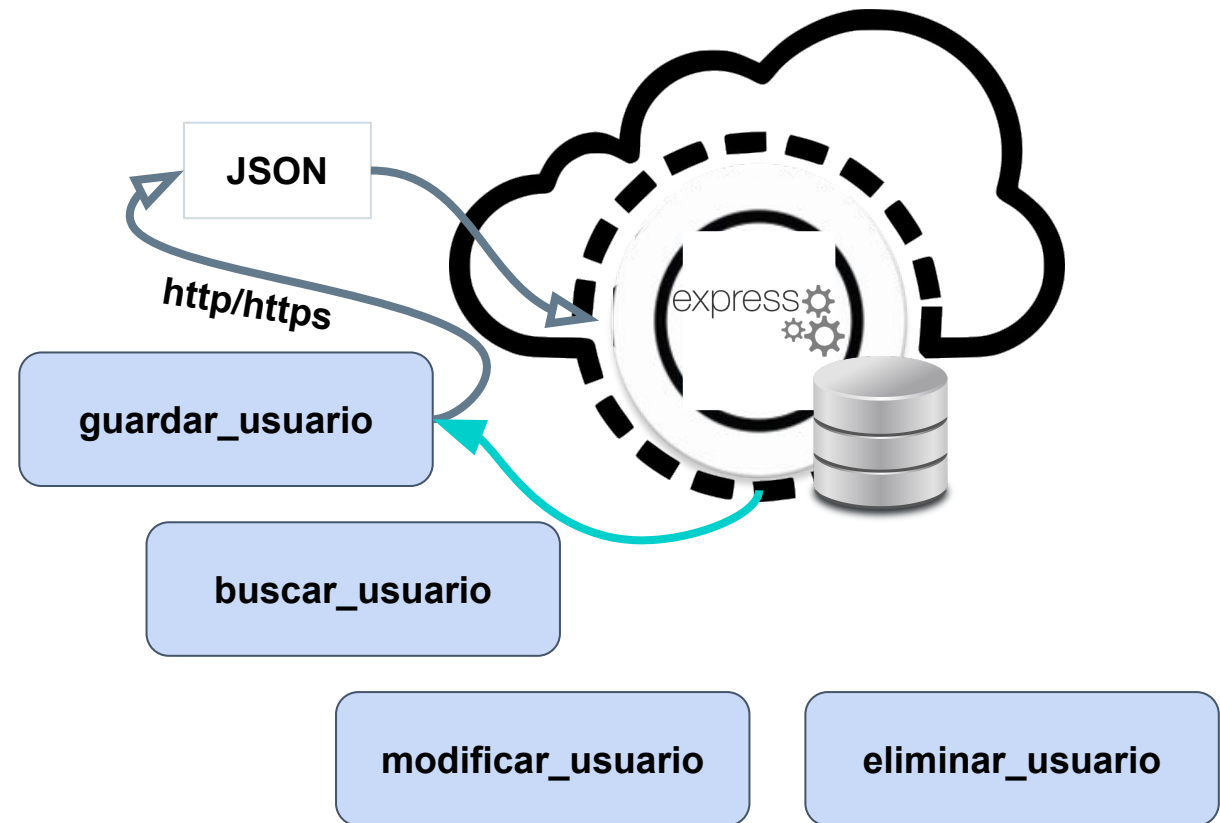
Arquitectura REST

REST

Representational State Transfer

Es **un estilo** de arquitectura de software que se creó para guiar el diseño y desarrollo de la arquitectura para la World Wide Web.

- Escalabilidad
- Interfaces uniformes
- Creación de servicios web



Desarrolla una API RESTful

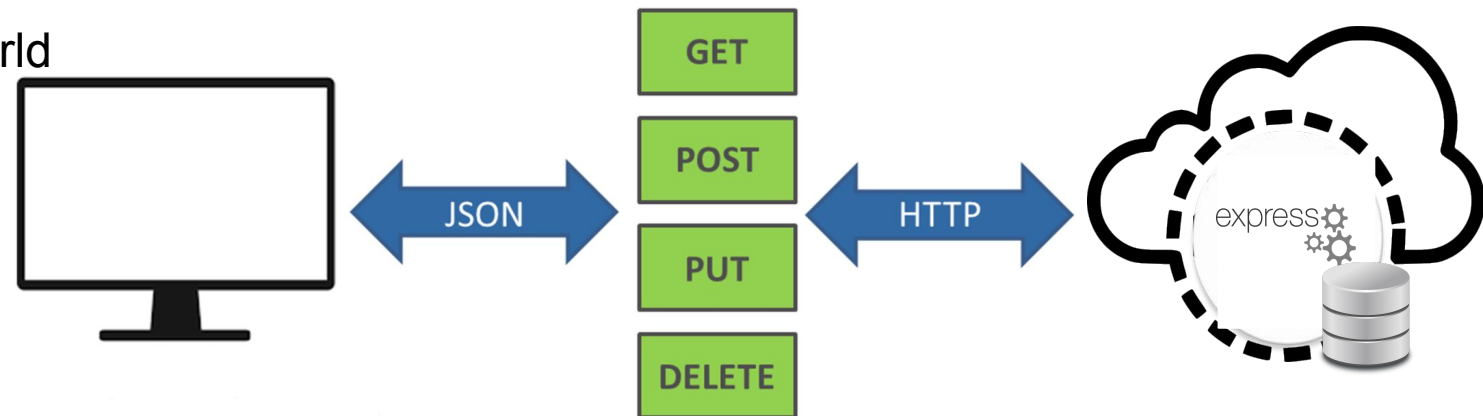
Arquitectura REST

REST

Representational State Transfer

Es **un estilo** de arquitectura de software que se creó para guiar el diseño y desarrollo de la arquitectura para la World Wide Web.

- Escalabilidad
- Interfaces uniformes
- Creación de servicios web



Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

<http://api.unservicioweb.com/usuarios>



Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

La URL en el contexto de microservicios se conoce como **endpoint**

<http://api.unservicioweb.com/usuarios>

protocolo

subdominio

dominio

Recurso

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

En los **endpoints** utiliza sustantivos para identificar los recursos

<http://api.unservicioweb.com/usuarios>

protocolo

subdominio

dominio

Recurso

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

El nombre de los recursos debe ser en plural

<http://api.unservicioweb.com/usuarios>

protocolo

subdominio

dominio

Recurso

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Método o verbo HTTP

GET

<http://api.unservicioweb.com/usuarios>

GET

<http://api.unservicioweb.com/eventos>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

GET

<http://api.unservicioweb.com/usuarios>

POST

<http://api.unservicioweb.com/usuarios>

GET

<http://api.unservicioweb.com/eventos>

POST

<http://api.unservicioweb.com/eventos>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Método o verbo HTTP

unservicioweb.com/**usuarios**

POST

http://api.unservicioweb.com/**usuarios**

GET

http://api.unservicioweb.com/**eventos**

POST

http://api.unservicioweb.com/**eventos**

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

GET

<http://api.unservicioweb.com/usuarios>

POST

<http://api.unservicioweb.com/usuarios>

PUT

<http://api.unservicioweb.com/usuarios/1>

DELETE

<http://api.unservicioweb.com/usuarios/3>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

GET

<http://api.unservicioweb.com/usuarios>

Método o verbo HTTP

<http://api.unservicioweb.com/usuarios>

PUT

<http://api.unservicioweb.com/usuarios/1>

DELETE

<http://api.unservicioweb.com/usuarios/3>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

A estos se les conoce como peticiones o solicitudes (**request**)

GET

<http://api.unservicioweb.com/usuarios>

GET

<http://api.unservicioweb.com/usuarios/1>

POST

<http://api.unservicioweb.com/usuarios>

PUT

<http://api.unservicioweb.com/usuarios/1>

DELETE

<http://api.unservicioweb.com/usuarios/3>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Por cada solicitud la aplicación servidor responde (**response**)

GET

<http://api.unservicioweb.com/usuarios>

GET

<http://api.unservicioweb.com/usuarios/1>

POST

<http://api.unservicioweb.com/usuarios>

PUT

<http://api.unservicioweb.com/usuarios/1>

DELETE

<http://api.unservicioweb.com/usuarios/3>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Por cada solicitud la aplicación servidor responde (**response**)

GET

<http://api.unservicioweb.com/usuarios>

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Por cada solicitud la aplicación servidor responde (**response**)

GET

<http://api.unservicioweb.com/usuarios>

Código de estado

200

```
{  
  "data": [ { ... }, { ... }, {  
    "id": "12",  
    "nombre": "Jefferson",  
    "apellido": "Peña",  
    "estado": "Activo"  
  }, { ... }, { ... } ]  
}
```

Desarrolla una API RESTful

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Por cada solicitud la aplicación servidor responde (**response**)

GET

<http://api.unservicioweb.com/usuarios/12>

Código de estado

200

```
{
  "data": {
    "id": "12",
    "nombre": "Jefferson",
    "apellido": "Peña",
    "estado": "Activo"
  }
}
```

Desarrolla una API **RESTful**

Sigue REST

RESTful = Implementa/Sigue REST

Un servicio que sigue REST puede tener varios niveles de madurez. Se considera RESTful si cumple con más de 2 de los niveles descritos por Richardson

Por cada solicitud la aplicación servidor responde (**response**)

1XX

INFORMACIÓN

2XX

ÉXITO

3XX

REDIRECCION

4XX

ERROR DEL CLIENTE

5XX

ERROR DEL SERVIDOR



www.upb.edu.co/es/mision-tic
#MisiónTICSomosTodos