Introducción a REST

Comprobar Versiones

- **Es aplicable para las siguientes versiones:**
  - ASP.NET Core 6.0.6
  - .NET 6.0
  - Visual Studio 2022

Comprobar Versiones

- **Es 100% aplicable a:**
  - ASP.NET Core 6.x
  - .NET 6.x

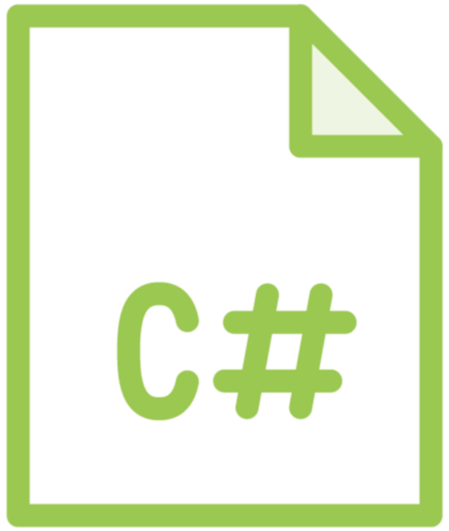
Resumen

- **Requisitos previos y herramientas**
- **ASP.NET Core MVC para crear APIS RESTful**
  - La aplicación de ejemplo
  - Introducción a Postman

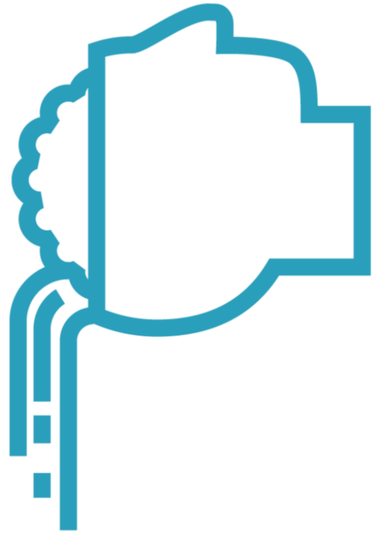
Resumen

- **Introducción a REST**
  - Aprendiendo sobre las restricciones de REST
- **El Modelo de Madurez de Richardson (Richardson Maturity Model)**

Requisitos previos del curso



Buenos conocimientos de C#



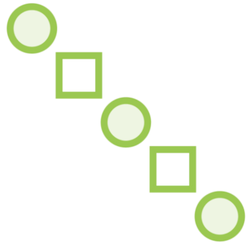
Algunos conocimientos de Asp.Net Core

Requisitos previos del curso

Aunque se repiten algunos conceptos importantes de la API, los aspectos básicos no se vuelven a tratar en este curso

Situar ASP.NET Core MVC para construir APIs RESTful

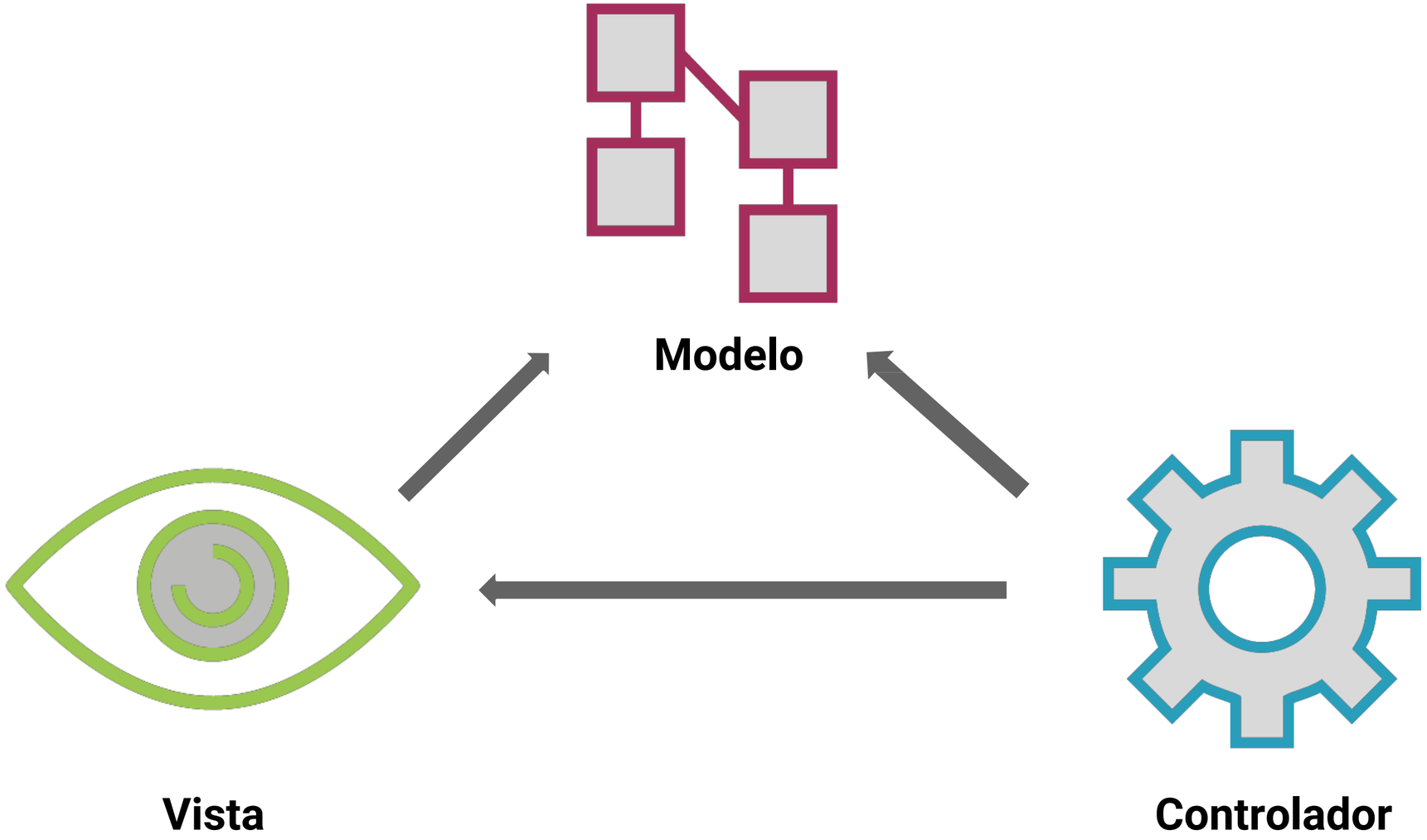
El Modelo-Vista-Controlador es un patrón arquitectónico orientado a implementar interfaces de usuario



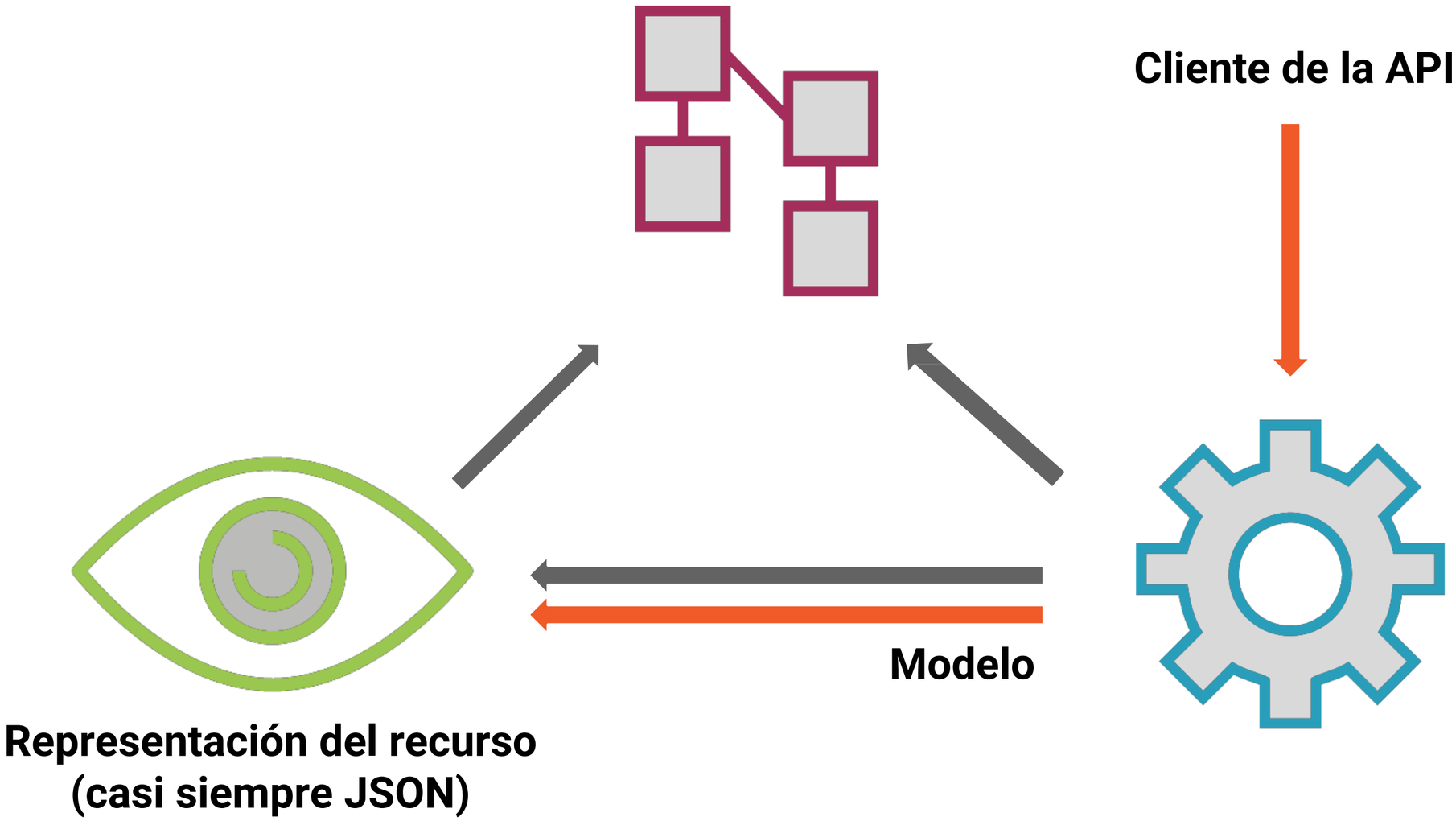
Fomenta el acoplamiento flexible y la separación de intereses

... ¡pero no es aplicable para toda la aplicación!

El patrón Modelo-Vista-Controlador (API)



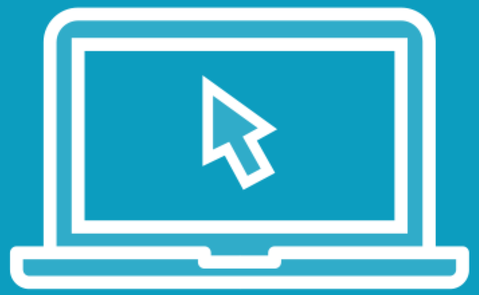
El patrón Modelo-Vista-Controlador (API)



No se obtiene una API RESTful de forma inmediata sólo porque se utilice el patrón MVC de ASP.NET Core

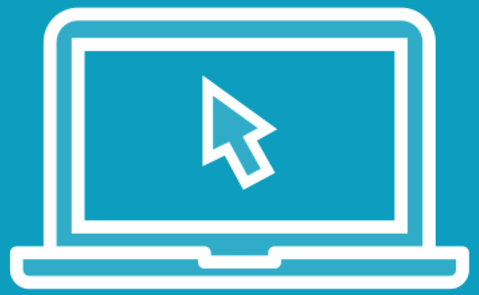
Esto se consigue cumpliendo una serie de restricciones RESTful que conoceremos

Demo



Presentación del proyecto inicial

Demo



Uso de Postman e importación de la colección con peticiones de ejemplo

Presentación de REST

- **REST es... ¿HTTP?**
- **REST es... ¿sobre la creación de APIs?**
- **REST es... ¿devolver JSON a partir de una petición HTTP?**

REST ...

Roy Fielding
https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

Representational State Transfer tiene por objeto describir el comportamiento de una aplicación web bien diseñada:

un conjunto de páginas web (una máquina virtual de estado) ...

... donde el usuario progresa a través de una aplicación seleccionando enlaces (transiciones de estado)...

... resultando en la siguiente página (representando el siguiente estado de la aplicación) siendo transferida al usuario y renderizada para su uso

Roy Fielding
https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

Introducción a REST

REST es un estilo arquitectónico

REST no es un estándar en sí mismo

Utilizamos estándares para implementar este estilo arquitectónico

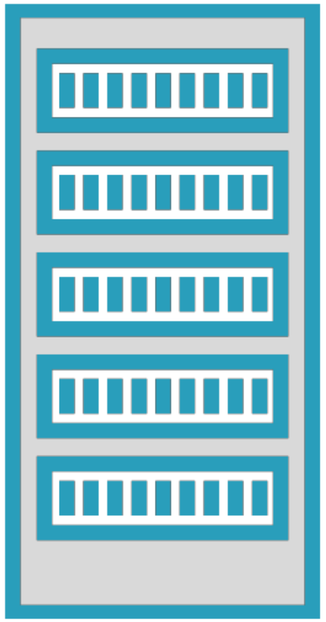
REST es agnóstico con respecto al protocolo

Representational State Transfer (REST)

php=>Tengo que saber ruta del archivo. Estático
ahora=>

HTTP request

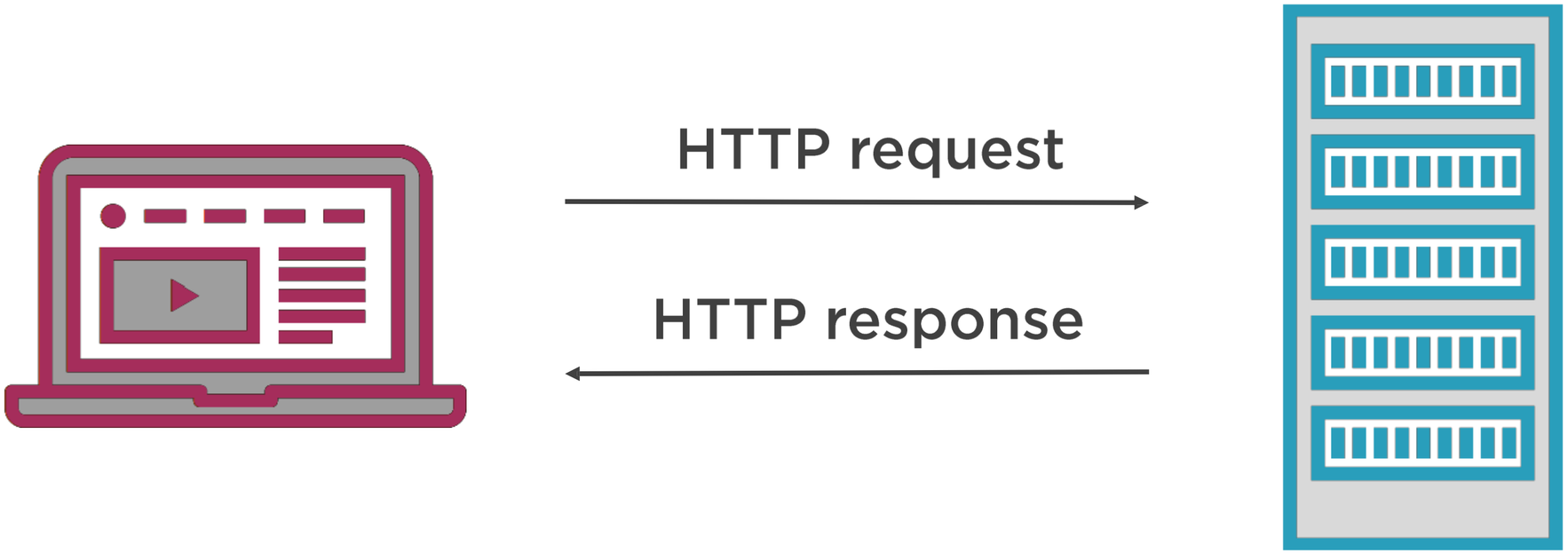
HTTP response



http://mynewspaper.com/index.html

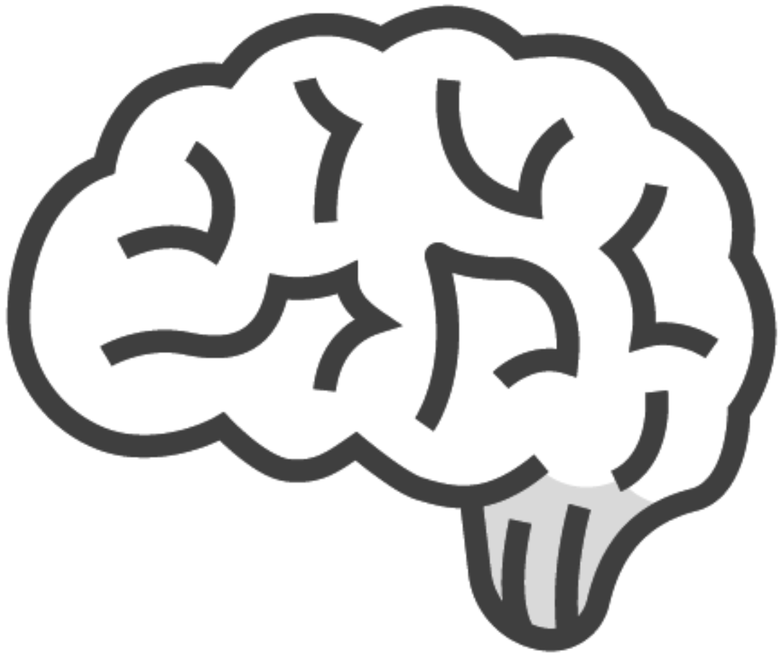
http://mynewspaper.com/article1.html

Representational State Transfer (REST)



http://myapi/authors/{id}/books
http://myapi/authors/{id}

Aprendiendo sobre las restricciones en REST



- **REST está determinado por 6 restricciones (una opcional)**
- **Una restricción es una decisión de diseño que puede tener impactos positivos y negativos**

Restricción

Es una decisión de diseño que puede tener impactos positivos y negativos

Aprendiendo sobre las restricciones en REST

Cliente-Servidor
El cliente y el servidor están separados (el cliente y el servidor pueden evolucionar por separado)

Sin estado
El estado está contenido en la solicitud

Cacheable
Cada respuesta debe indicar explícitamente si puede ser almacenada en caché o no

Aprendiendo sobre las restricciones en REST

Sistema de capas
El cliente no puede saber a qué capa está conectado

Código bajo demanda (opcional)
El servidor puede ampliar la funcionalidad del cliente

Interfaz Uniforme
API y los consumidores comparten una única interfaz técnica: URI, método, tipo de medio

Un sistema sólo se considera restful cuando se adhiere a todas las restricciones requeridas

la mayor parte de las APIs " RESTful " no son realmente RESTful ....
... pero eso no les convierte en malas apis, siempre y cuando se entiendan las posibles contrapartidas.

El modelo de madurez de Richardson

Este modelo clasifica las APIs según su madurez RESTful

El modelo de madurez de Richardson

Nivel 0 – Swamp of POX

El protocolo HTTP se utiliza para interacción remota.

… el resto del protocolo no se utiliza como es debido.

Implementaciones de estilo RPC (SOAP, visto muchas veces al utilizar WCF)

```
POST (info on data)
http://host/myapi
```

```
POST (author to create)
http://host/myapi
```

El modelo de madurez de Richardson

Nivel 1 – Recursos

Cada recurso se mapea a una URI

Los métodos HTTP no se utilizan tal como debieran ser utilizados.

Se obtiene un resultado menos complejo

```
POST
http://host/api/authors
```

```
POST
http://host/api/authors/{id}
```

El modelo de madurez de Richardson

aquí estamos!!!

Nivel 2 – Verbos HTTP

Se utilizan los verbos HTTP adecuados

Se utilizan los códigos de estado adecuados

Elimina las variantes innecesarias

```
GET
http://host/api/authors
200 Ok (authors)


POST (author representation)
http://host/api/authors
201 Created (author)
```

El modelo de madurez de Richardson

vamos a aprender esto

Nivel 3 – Hypermedia

La API soporta HATEOAS - Hypermedia as the Engine of Application State (Hipertexto como el mecanismo de estado de la aplicación)

Permite el descubrimiento de servicios

```
GET
http://host/api/authors
200 Ok (authors + links that
drive application state)
```

El modelo de madurez de Richardson

El nivel 3 es un requisito previo para obtener una API RESTful

Resumen

ASP.NET Core MVC proporciona un marco de trabajo para construir APIs y aplicaciones web utilizando el patrón Modelo-Vista-Controlador

Resumen

- REST es un estilo arquitectónico que representa una imagen de cómo debe comportarse una aplicación web bien diseñada
- Seis restricciones
  - Interfaz Uniforme
  - Cliente-Servidor
  - Sin Estado
  - Sistema de capas
  - Cacheable
  - Código bajo demanda

Resumen

- El modelo de madurez de Richardson clasifica las APIs por su madurez en cuanto a RESTful
  - El nivel 3 es un requisito previo para obtener una API RESTful

Siguiente:
Diseño del contrato exterior