AJAX y ASINCRONISMO

# Parte II - AJAX, Asincronismo, REST

1. ¿Qué es y en qué consiste una API?

Una API es una interfaz que nos da una aplicación para comunicarnos con ella.

1. ¿Qué es un servicio REST ? REpresentational State Tranfer

Es un tipo de arquitectura de desarrollo web y es un estándar para crear APIs para servicios orientados a internet. La mayoría (no todas) de las APIs Rest usan Json para comunicarse.

JSON = JavaScript Object Notation

Presenta un formato ligero para el intercambio de datos y es una alternativa a XML como representación de objetos. Ejemplo :

let objeto =

{

"propiedad": valor,

"propiedad2": valor2,

"arreglo": [val1, val2]

}

* Describa como se transfieren los datos y sobre que protocolo.

Se asocian URLs a recursos al que se puede acceder o modificar mediante los métodos del protocolo HTTP.

* Enuncie las acciones que permiten manipular los datos y expique que hace cada una.

Se basa en acciones (llamadas verbos) que manipulan los datos.

POST: Crear un recurso

GET: Obtener uno o muchos recursos

PUT: Actualizar uno o muchos recursos

DELETE: Borrar un recurso.

GET /facturas (en genérico /facturas)

Acceder al listado de facturas

POST /facturas (en genérico /facturas)

Crear una factura nueva

GET /facturas/123 (en genérico /facturas/:id\_fact)

Acceder al detalle de una factura

PUT /facturas/123 (en genérico /facturas/:id\_fact)

Editar la factura, sustituyendo la totalidad de la información anterior por la nueva.

DELETE /facturas/123 (en genérico /facturas/:id\_fact)

Eliminar la factura

* Mencione errores que pueden suceder.

Se utilizan los errores del protocolo HTTP.

200 ok, 404 not found, etc.

Se pueden utilizar los errores del protocolo HTTP:

200 OK Standard response for successful HTTP requests

201 Created

202 Accepted

301 Moved Permanently

400 Bad Request

401 Unauthorised

402 Payment Required

403 Forbidden

404 Not Found

405 Method Not Allowed

500 Internal Server Error

501 Not Implemented

## **BÁSICO**

3. Acceder a las siguientes URL’s:

1. <https://jsonplaceholder.typicode.com/posts>
2. <https://web-unicen.herokuapp.com/api/groups/ejemplos/nombres>

¿Qué es lo que muestra y por qué? Analice la estructura de los datos. Puede utilizar [este plugin](https://chrome.google.com/webstore/detail/json-formatter/bcjindcccaagfpapjjmafapmmgkkhgoa?hl=es) para ver mejor la respuesta.

a. Devuelve un arreglo con objetos json [

{

"userId": 1,

"id": 1,

"title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",

"body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"

},…]

Cada objeto es tiene la forma {

"userId":userId,

"id":id,

"title":title,

"body":body}

b.Devuelve un json con el formato

{“status”:status,

“nombres”[

{“\_id”:\_id, “group”:group, “thingtype”:thingtype, “thing”:{“nombre”:nombre}, “dateAdded”:dateadded, “\_\_v”:\_\_v }

,…]}

4. Si se desean recibir y enviar datos a un servicio, analizar qué se debe definir en el parámetro *opciones* de la interfaz **fetch**:

fetch(url, opciones).then(...)

5. Obtenga mediante un llamado ajax los datos están en el **servicio a y b del punto 3** y muestrelos en una lista de una página html.

6. Implementar un envío de datos al servicio de nombres de la cátedra: <https://web-unicen.herokuapp.com/api/groups/ejemplos/nombres>.

Considere:

* Debe llenar a través de un input el nombre que quiere enviar
* Debe mostrar en la página el dato enviado, el JSON enviado y la url donde envía.
* Una vez enviado, vuelta a mostrar la lista usando la función que hizo en el ejercicio 5.

## **INTERMEDIO**

7. Borre algun item específico del servicio de nombres.

8. Realice la modificación de alguno de los datos del servicio a través de un formulario donde especifique el ID y el nombre nuevo. Muestre en la página el dato antiguo, el dato nuevo y verifique que fue modificado.

**AVANZADO**

10. Desde los datos de prueba de la API de clima OpenWeather, obtenga los datos de temperatura, presión y humedad para la ciudad de Tandil.

Para poder utilizar la API es necesario obtener una API KEY siguiendo estos pasos: <https://openweathermap.org/appid>. Además debe investigar el formato de los llamados siguiendo la documentación oficial: <https://openweathermap.org/current>

* Muéstre los datos en una tabla que contenga como encabezado, Temperatura, Presión, y Humedad.
* Exprese la temperatura en grados centígrados (por defecto está en grados Kelvin).
* Muestre una imagen que dependa de la temperatura. Ej: Si supera cierta temperatura un sol.

11. Crear una aplicación web que permita:

* Cargar en un servicio mediante un formulario datos de alumnos de una asignatura. Nombre, Apellido, DNI y 3 Notas de Exámenes.
* Contar los aprobados / desaprobados, y mostrar porcentaje de aprobación.
* Calcular y mostrar la Nota media de cada examen.
* Mostrar un listado de alumnos ordenado por apellido.
* Mostrar en una tabla todos los alumnos con sus datos y su respectivo promedio.
* Borrar alumnos localizados por nombre y apellido través de un input.

12. Investigue y utilice la plataforma <https://my-json-server.typicode.com/> para crear “Fake Rest Services”.

* Utilice alguno de los recursos “demo”
* Cree sus propios recursos