
Desarrollo del lado servidor con NodeJS, Express y MongoDB

APIs y Webs

El desarrollo web seguramente te resulte familiar. Hoy en día hay miles de proyectos webs que viven en la nube y permiten el acceso desde cualquier parte del mundo a sus usuarios. Se utilizan herramientas de programación web como ser javascripts, .net, rails y miles más. Para la persistencia de los datos, utilizamos justamente bases de datos. Las hay de varios tipos, orientadas a objetos, documentos, relacionales, no relacionales, de grafos, etc. Todas sirven al propósito de la persistencia y elegiremos la adecuada de acuerdo a las necesidades del proyecto o módulo que estemos creando.

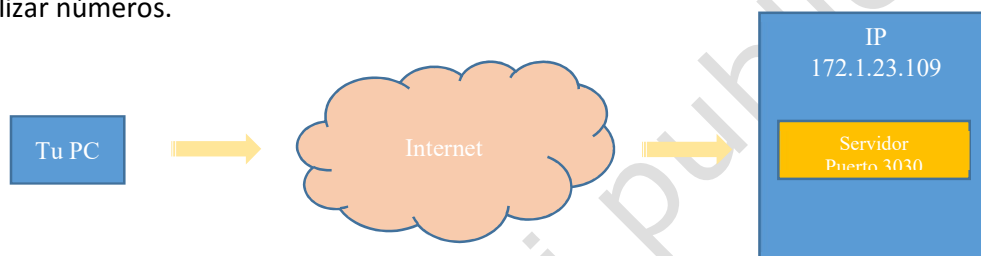
¿Cómo es la estructura básica de un proyecto web?

Este no es un curso de redes de comunicaciones, pero es importante entender los conceptos básicos de un proyecto web, a fin de que hablemos el mismo idioma y no te resulten extraños algunos conceptos.

Tenemos el servidor que se encuentra dentro de una máquina. Para acceder a él se debe permitir escuchar conexiones remotas y, para eso, el usuario debe conocer su IP y puerto, que hacen las veces de dirección única. Así como se puede identificar tu domicilio brindando una calle, número y demás, en el mundo web necesitamos el número IP, para llegar al servidor y al puerto para acceder.

La estructura de una IP es: xxx.xxx.xxx.xxx (ej: 168.1.40.10). Hoy en día ya existen las IP V6 que agregan un par más de números de identificaciones para ampliar la cantidad de direcciones disponibles. Puedes entrar al sitio <https://www.whatismyip.com/es/> y confirmar tu IP. Todo dispositivo conectado a internet tiene una IP.

Quizá te estés preguntando, si todo tiene una IP y es la única forma de acceder a un dispositivo conectado a internet, ¿por qué nunca lo has usado? Eso es porque usas un nombre de dominio público que hace las veces de alias de una IP. Concretamente, al crear un nombre de dominio, por ejemplo, www.google.com, se lo debe configurar asociándolo con una IP. Hay varios pasos que estoy omitiendo, pero el objetivo es que comprendas como se accede a un servidor web y no tanto su configuración a nivel de redes. Desde aquí <https://www.whatismyip.com/dns-lookup/> puedes ingresar una URL y ver su IP vinculada. De esta manera, podemos recordar sitios fácilmente en vez de utilizar números.



Ahora bien, la computadora conectada a internet puede estar ofreciendo diferentes servicios, ¿cómo identificamos el que nos interesa? Eso se hace a través del número de puerto. Entra a la siguiente sección <https://www.whatismyip.com/port-scanner/> y elige el *package* web para ver los servicios que estás exponiendo. De esta forma, una computadora puede exponer servicios al mundo a través del IP y puerto. Cuando instales un servidor web, vas a definir un puerto que escuche las conexiones entrantes y responda de acuerdo a lo que se solicite, una página web o bien el resultado de una operación sobre nuestra API.

Para acceder a una página web, utilizaremos el protocolo HTTP¹. De esta forma, vía http podremos solicitar una página a un servidor y verla desde un explorador como Chrome, Safari, Mozilla u otros.

¹ https://es.wikipedia.org/wiki/Protocolo_de_transferencia_de_hipertexto

² <https://developer.android.com/reference/>

¿Qué sucede con las APIs?

Antes que nada, entendamos qué es una API. Por definición, una API significa *Application Program Interface*, o bien, interfaz de software para interactuar con otro software. Pensemos en un producto, como puede ser un acondicionador de aire. Tiene capacidad para enfriar un ambiente pero, ¿cómo interactúas con esto? Tienes botones que controlan la temperatura, inclinación de las salidas de ventilización y un reloj para programar encendido y apagado del dispositivo, entre otras cosas. Estos controles son la interfaz de usuario que nos permiten interactuar con el aparato.

Aun sin saber cómo logra realizar todas esas tareas para enfriar el ambiente, sí sabemos cómo indicarle que lo haga. A esto lo llamamos una abstracción del dispositivo que nos permite lidiar con él sin necesidad de conocer cómo funciona. Podemos pensarlo como que la interfaz de usuario responde al **qué** y la implementación al **cómo**. Lo mismo sucede con los componentes de software que nos brindan una abstracción de su uso, o interfaz, o simplemente API. Por ejemplo, cuando uno programa en Android² o iOS, utiliza APIs que proveen los desarrolladores del sistema operativo para acceder a distintas funciones del teléfono o tableta.

¿Qué son las Web APIs?

Sencillamente son API que se operan mediante el protocolo HTTP. La ventaja de operar vía web, es que el que requiera utilizar esa API, puede hacerlo desde cualquier lenguaje que utilice el protocolo HTTP. Por lo tanto, es una enorme ventaja al momento de conectar dispositivos, software, o lo que queramos a internet.

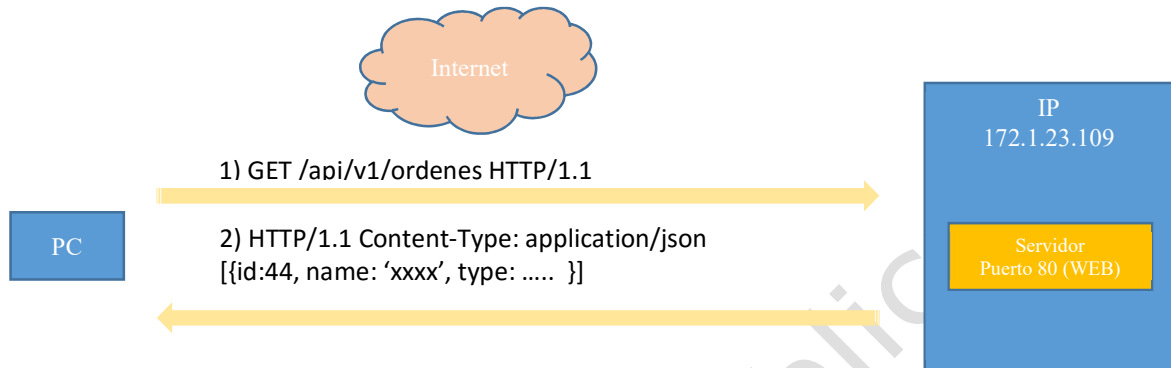
Te animo a que busques las API de Facebook, Twitter, Github para que veas ejemplos de *Web APIs* y sus usos.

¹ https://es.wikipedia.org/wiki/Protocolo_de_transferencia_de_hipertexto

² <https://developer.android.com/reference/>

¿Cómo funcionan?

Veamos un ejemplo utilizando la siguiente figura:



Lo primero que se realiza es la solicitud vía HTTP a tracy.com con el verbo GET a la url api/v1/ordenes. Eso llega al servidor que tiene la IP 172.1.23.109 y entra por el puerto 80 que está vinculado a la web. Procesa el requerimiento y retorna un json.

A la url tracy.com/api/v1/ordenes se le llama **endpoint**.

Nos detendremos aquí, ya que luego profundizaremos en estos temas.

Espero que hayas comprendido cómo funcionan las APIs, para qué sirven, similitudes con los proyectos web y diferentes usos posibles.

En este punto del curso, lo más importante es que te imagines el potencial que tenemos al desarrollar una API y exponerla al mundo.

Pasemos a conocer Node JS.