# Trabajo Práctico

## EJERCICIO 2

1. Un servidor HTTP es un servicio que se encarga de manejar las solicitudes que el cliente realiza, este es encargado de procesar esos pedidos y enviar una respuesta al cliente que la solicito.
2. El protocolo de comunicación HTTP posee diferentes tipos de verbos con los cuales el cliente puede comunicar al servidor que es lo que desea hacer con su petición, ya que estas pueden ser por ejemplo de lectura de información o de escritura.

Algunos de los verbos más utilizados en este protocolo son:

* GET:

El método GET sirve para solicitar al servidor información.

* POST:

El método POST se utiliza para enviar información al servidor.

* PUT:

El modo PUT reemplaza la información actual del recurso solicitado en la petición.

* DELETE

El método DELETE borra el recurso que fue solicitado en la petición.

1. El request y el response como se mencionó anteriormente son respectivamente las consultas que el cliente hace al servidor y la respuesta que este brinda al cliente.

Una consulta está compuesta por 3 partes. Primeramente, una línea que lleva el verbo el recurso al que se desea acceder y luego la versión de http que debe ser utilizada, a esto se lo llama request line.

Luego tenemos los headers o encabezados que se encargan de brindar información adicional tanto de la petición como de la respuesta como puede ser el header host para indicar a que dominio debemos hacer la petición.

Por último, se encuentra el body que es una parte opcional del mensaje y puede ser utilizado por ejemplo en solicitudes POST donde debemos colocar en body la información que deseamos enviar al servidor.

1. Una queryString en una URL es una cadena de texto que permite hacer búsquedas específicas para que a la hora de manejar una solicitud esta información llegue al servidor donde podrá ser procesada y así devolver solo la información que cumpla con los parámetros enviados por la queryString.
2. Un responseCode es uno de los elementos de un mensaje de respuesta del servidor luego de una solicitud, se encuentra en la primera línea del mensaje e indica el estado de la respuesta es decir si esa solicitud fue realizada con éxito o si hubo algún tipo error, los códigos se encuentran clasificados en la siguiente lista

* Respuestas informativas (100–199),
* Respuestas satisfactorias (200–299),
* Redirecciones (300–399),
* Errores del cliente (400–499),
* y errores del servidor (500–599).

1. Los verbos GET y POST tienen formas diferentes de enviar la información solicitada ya qué tienen funciones distintas, el verbo GET envía la información mediante la URL, por lo que es visible para todos incluso queda registrada en el historial de búsqueda, en cambio la información enviada a través de POST viaja de forma protegida y no es visible en la URL.

Por otro lado, la cantidad de datos que podemos enviar mediante una solicitud GET está limitada al largo máximo de la URL mientras que con POST la cantidad de datos para enviar no tiene límite.

1. Cuando abrimos una página web lo primero que hace nuestro navegador es enviar una request al servidor al que estamos conectando mediante un GET para que este retorne los archivos necesarios para mostrar la página web solicitada.

Esto podemos verlo al abrir la consola del desarrollador de nuestro explorador y ver los request realizados por nuestro cliente al servidor que aloja la página que intentamos abrir.

1. Tanto JSON (JavaScript Object Notation) como XML (Extensible Markup Languaje) son formatos de textos simples que permiten el traspaso de información rápidamente mediante determinadas estructuras, por ejemplo.

En JSON:

**{**

**"persona”: {**

**"nombre\_atributo": "valor",**

**"nombre": "Agustin",**

**"edad": 20**

**"idiomas”: {**

**"español":"nativo",**

**"portugués":"intermedio",**

**"ingles": "avanzado"**

**}**

**}**

**}**

El equivalente de este objeto en XML

**<persona>**

**<nombre\_atributo> valor </nombre\_atributo>**

**<nombre>Agustin</nombre>**

**<edad>20</edad>**

**<idiomas>**

**<español>nativo</español>**

**<portugués>intermedio</portugués>**

**<ingles>avanzado</ingles>**

**</idiomas>**

**</persona>**

1. SOAP es un estándar de intercambio de información que permite la comunicación entre elementos mediante texto en formato XML. SOAP es flexible ya que permite la comunicación mediante cualquier protocolo ya sea HTTP, TCP o SMTP. Sin embargo, SOAP puede ser considerablemente más lento que otros estándares.
2. REST Surge como una alternativa a SOAP brindando una solución más sencilla y rápida de usar mediante el protocolo HTTP, esta solución permite una separación clara entre el cliente y el servidor lo que lo hace escalable y mucho más flexible a la hora de desarrollar aplicaciones.
3. El contentType en un mensaje HTTP es una propiedad del header utilizado para indicar el tipo de dato del recurso que el servidor está enviando como respuesta puede ser por ejemplo "image/png" o "text/html"