



INSTITUTO TECNOLÓGICO SUPERIOR DE JEREZ



INGENIERÍA EN SISTEMAS COMPUTACIONALES

PROGRAMACIÓN WEB

9° SEMESTRE

TEMA 2: HTML, XML y CSS

M.T.I. SALVADOR ACEVEDO SANDOVAL

“MAPA CONCEPTUAL”

DEISY LOZANO GARCÍA

No. Control: S16070121

Correo: deisy10logar@hotmail.com

**JEREZ, ZACATECAS
VIERNES 25 DE SEPTIEMBRE DE 2020**

Cuestionario

1. ¿Qué es el protocolo HTTP y para qué sirve?

El protocolo HTTP (HyperText Transfer Protocol) es un protocolo simple de tipo solicitud-respuesta incluido dentro de la familia de protocolos TCP/IP que se utiliza en Internet. Esto quiere decir que, cada vez que accedemos a una página (en general, a un recurso accesible a través de HTTP), se establece una conexión diferente e independiente de las anteriores.

Funcionamiento:

Cada solicitud del cliente tiene como resultado una respuesta del servidor y, cada vez que el cliente hace una solicitud, ésta se realiza de forma independiente a las anteriores.

(Berzal, Cortijo, & Cubero)

Características:

- + Es un protocolo cliente – servidor que se emplea para el intercambio de información de la web entre ellos.
- + El protocolo HTTP, utiliza el puerto definido 80.
- + Es un protocolo de aplicación que funciona sobre los servicios de TCP/IP

(Valdivia Miranda, 2017)

- + Sirve para páginas web.
- + Es utilizado para solicitar y transmitir archivos a través de internet u otra red informática, especialmente páginas web.
- + Está orientado a transacciones.
- + Opera a través de un esquema petición – respuesta, entre un cliente y un servidor.

(Ramos Martín & Ramos Martín, 2011)

2. ¿Cuál es la versión más actual de dicho protocolo?

La versión más reciente es HTTP/2:

- ✚ Es un protocolo binario, es posible utilizar en él técnicas de optimización.
- ✚ Es un protocolo multiplexado, es decir, se pueden hacer peticiones paralelas sobre la misma conexión.
- ✚ Comprime las cabeceras, lo que realiza, es la eliminación de la duplicación y retardo en los datos a transmitir.

(Evolución del protocolo HTTP, 2019)

3. ¿Qué información tiene un encabezado HTTP?

Las cabeceras HTTP permiten al cliente y al servidor enviar información adicional junto a una petición o respuesta.

Una cabecera de petición está compuesta por su nombre, seguido de dos puntos y a continuación su valor.

```
[nombre del encabezado]: [valor del encabezado]
```

Las cabeceras pueden ser agrupadas de acuerdo a sus contextos:

✚ **Cabecera General:**

Se aplican tanto a las peticiones como a las respuestas, pero sin relación con los datos que finalmente se transmiten en el cuerpo.

✚ **Cabecera de Consulta:**

Contienen más información sobre el contenido que va a obtenerse o sobre el cliente.

✚ **Cabecera de Respuesta:**

Contiene más información sobre el contenido, como su origen o el servidor.

✚ **Cabecera de Entidad:**

Contiene más información sobre el cuerpo de la entidad, como el tamaño del contenido o su tipo MIME.

(HTTP headers, 2019)

4. ¿Qué información muestra un Código de Estado en una respuesta HTTP?

Los códigos de estado se devuelven en respuestas HTTP a solicitudes realizadas a la Pasarela Web de WebSphere MQ Managed File Transfer.

La cabecera de una respuesta devuelta por la Pasarela web contiene un código de respuesta HTTP. La cabecera HTTP en el siguiente ejemplo contiene el código de respuesta HTTP 200 OK.

```
HTTP/1.1 200 OK
Server: WAS/6.0
Content-length: 0
```

(Códigos de respuesta HTTP, s.f.)

5. ¿Cuáles son los métodos de comunicación de este protocolo?

Se definen ocho de los denominados métodos, los cuales indican la acción que se desea que sea realizada sobre el recurso identificado.

Los métodos definidos son los siguientes:

 **GET:**

Solicita una representación del recurso especificado.

 **HEAD:**

Pide una respuesta idéntica a la que correspondería a una petición GET, pero sin el cuerpo de respuesta.

Es útil para recuperar la información escrita en cabeceras de respuestas, sin necesidad de llevar el contenido entero.

POST:

Envía datos para ser procesados al recurso identificado.

Los datos se incluyen en el cuerpo de la petición.

Esto puede causar la creación de un nuevo recurso y/o las actualizaciones de recursos existentes.

Puede usarse para enviar a un servidor web los datos de un formulario.

PUT:

Carga una representación del recurso especificado.

DELETE:

Suprime el recurso especificado.

TRACE:

Provoca que el servidor devuelva en la respuesta la petición recibida.

Se suele usar para tareas de diagnóstico.

OPTIONS:

Devuelve los métodos HTTP que el servidor suministra para un URL especificado.

CONNECT:

Convierte la conexión de solicitud en un túnel transparente TCP/IP, por lo general para facilitar la comunicación SSL cifrada a través de un proxy HTTP no encriptado.

(Sivianes Castillo, y otros, 2010)

6. ¿Qué es el protocolo HTTPS y para qué sirve?

El protocolo seguro de Transferencia de Hipertexto (HTTPS, Hiper Text Transfer Protocol Secure) es la versión segura del protocolo HTTP. La diferencia es que HTTPS permite realizar transacciones de forma segura. Por lo tanto, podremos

desarrollar actividades de tipo e-commerce, acceso a cuentas bancarias on line, tramites con la administración pública, etc.

(Ferrer Martínez, Septiembre, 2014)

Características:

- Utiliza el cifrado basado en SSL/TTL.
- Utiliza técnicas criptográficas de clave pública para proteger los datos transmitidos y que sólo el destinatario pueda leerlos. Estas técnicas consisten en utilizar pares de claves emitidas por autoridades de certificación.

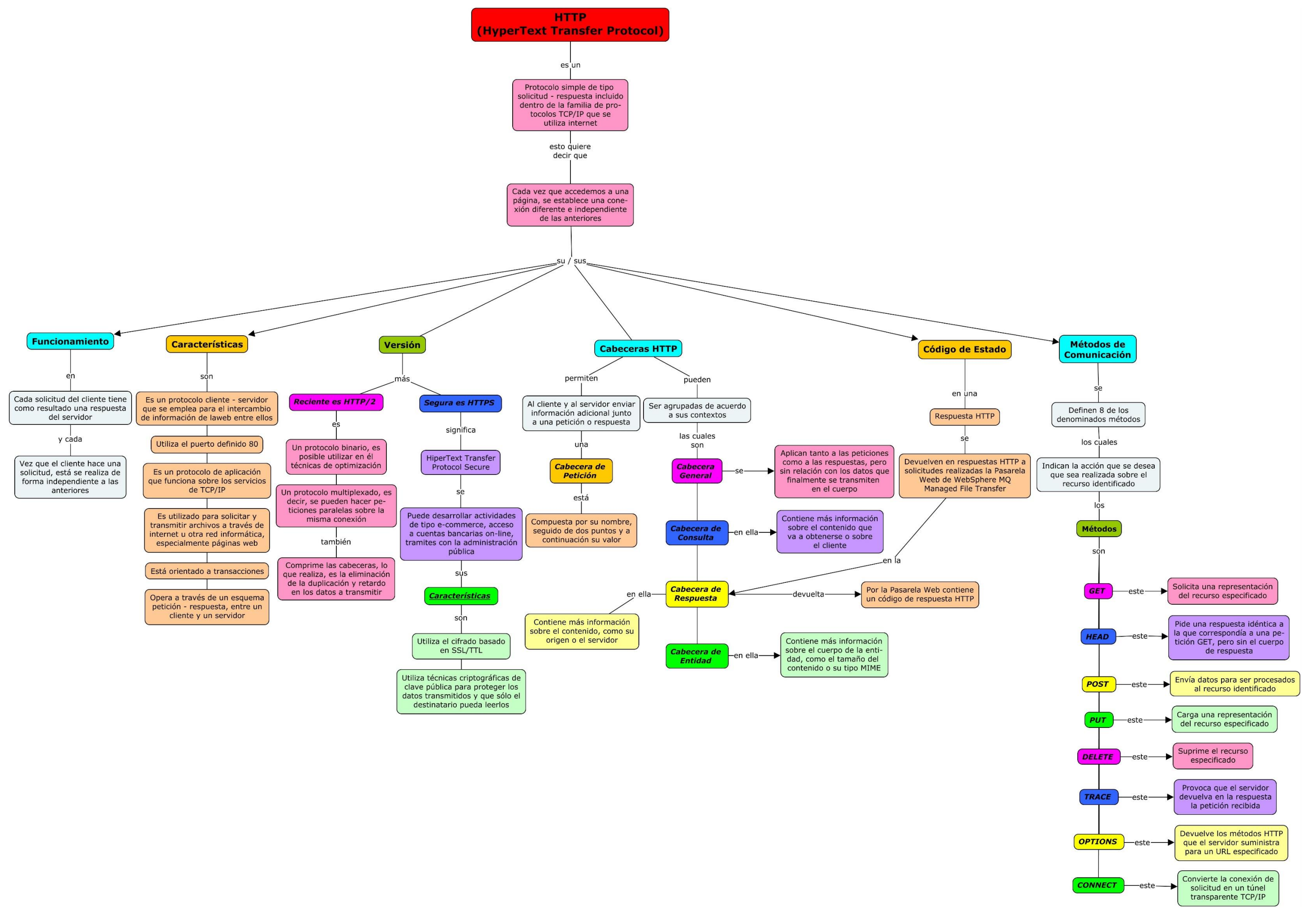
(Berzal, Cortijo, & Cubero)

7. Códigos de error de una respuesta HTTP.

Código De Respuesta HTTP	Ejemplo de Código de Error de WebSphere MQ Managed File Transfer
200 OK	-
202 Accepted	-
400 Bad Request	BFGWI0001
403 Forbidden	BFGWI0056
404 Not Found	BFGWI0015
405 Method Not Allowed	BFGWI0016
410 Resource Gone	BFGWI0031
413 Request Entity Too Large	BFGWI0026
415 Unsupported Media Type	BFGWI0017
500 Internal Server Error	BFGWI0018
502 Bad Gateway	BFGWI0019
503 Service Unavailable	BFGWI0020
504 Gateway Timeout	BFGWI0021

(Códigos de respuesta HTTP, s.f.)

Mapa Conceptual: HTTP



Referencias

Berzal, F., Cortijo, F. J., & Cubero, J. C. (s.f.). *Desarrollo Profesional de Aplicaciones Web con ASP.NET*. ISBN 84-609-4245-7. Recuperado el 23 de Septiembre de 2020, de https://books.google.com.mx/books?id=J1d_9l6zIAIC&printsec=frontcover#v=onepage&q&f=false

Códigos de respuesta HTTP. (s.f.). Recuperado el 23 de Septiembre de 2020, de ibm: https://www.ibm.com/support/knowledgecenter/es/SSFKSJ_7.5.0/com.ibm.wmqfte.doc/web_httpresponsecodes.htm

Evolución del protocolo HTTP. (23 de Marzo de 2019). Recuperado el 23 de Septiembre de 2020, de [developer.mozilla.org](https://developer.mozilla.org/es/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP): https://developer.mozilla.org/es/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP

Ferrer Martínez, J. (Septiembre, 2014). *Implantación de aplicaciones web (GRADO SUP.)*. Madrid, España: RA-MA. Recuperado el 23 de Septiembre de 2020, de <https://books.google.com.mx/books?id=eo-fDwAAQBAJ&printsec=frontcover#v=onepage&q&f=false>

HTTP headers. (25 de Agosto de 2019). Recuperado el 23 de Septiembre de 2020, de MDN web docs: <https://developer.mozilla.org/es/docs/Web/HTTP/Headers>

Ramos Martín, A., & Ramos Martín, M. J. (2011). *Aplicaciones Web* (Primera ed.). España: PARANINFO. Recuperado el 23 de Septiembre de 2020, de <https://books.google.com.mx/books?id=LXs3YIMoeNgC&printsec=frontcover&hl=es#v=onepage&q&f=false>

Sivianes Castillo, F., Sánchez Antón, G., Ropero Rodríguez, J., Rivera Romero, O., Benjumea Nodéjar, J., Barbancho Concejero, J., & Romero Ternero, M. D. (2010). *Servicios en Red* (Primera ed.). Madrid, España: PARANINFO.

Recuperado el 23 de Septiembre de 2020, de
<https://books.google.com.mx/books?id=aOb3rTabO44C&printsec=frontcover&hl=es#v=onepage&q&f=false>

Valdivia Miranda, C. (2017). *Informática Industrial* (Primera ed.). España: PARANINFO. Recuperado el 23 de Septiembre de 2020, de
<https://books.google.com.mx/books?id=heOMDgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>