

# Instituto Politécnico Nacional

## Escuela Superior de Cómputo

Web App Development.

Tarea 2 : Códigos de estado de respuesta HTTP

*Profesor: M. en C. José Asunción Enríquez Zárate*

*Alumno: Mauro Sampayo Hernández*

*mauro\_luigi@hotmail.com*

*3CM18*

28 de octubre de 2021

# Índice

<b>1. Introducción</b>	<b>1</b>
<b>2. Desarrollo</b>	<b>2</b>
2.1. Clasificación de los códigos de respuesta del protocolo HTTP. . . . .	2
2.2. Códigos de respuesta del protocolo HTTP. . . . .	2
2.2.1. Respuestas informativas (100 - 199): . . . . .	2
2.2.2. Respuestas satisfactorias (200 - 299): . . . . .	2
2.2.3. Redirecciones (300 - 399): . . . . .	3
2.2.4. Errores de los clientes (400 - 499): . . . . .	4
2.2.5. Errores de los servidores (500 - 599): . . . . .	6
<b>3. Conclusión</b>	<b>7</b>
<b>4. Referencias Bibliográficas</b>	<b>8</b>

## 1. Introducción

Los códigos de estado HTTP son mensajes que describen de forma abreviada la respuesta HTTP, e indican si se ha completado satisfactoriamente una solicitud HTTP específica. Este tipo de mensajes se devuelven cada vez que un navegador interactúa con un servidor y son una herramienta invaluable para diagnosticar y arreglar errores de configuración del sitio web.

El primer dígito del código de estado especifica uno de los 5 tipos de respuesta, el mínimo para que un cliente pueda trabajar con HTTP es que reconozca estas 5 clases. La Internet Assigned Numbers Authority (IANA) mantiene el registro oficial de códigos de estado HTTP.

## 2. Desarrollo

### 2.1. Clasificación de los códigos de respuesta del protocolo HTTP.

Los códigos de estado HTTP se dividen en 5 clases:

- **Respuestas informativas (100 - 199):** Códigos informativos que indican que la solicitud iniciada por el navegador continúa.
- **Respuestas satisfactorias (200 - 299):** Códigos devueltos cuando la solicitud del navegador fue recibida, entendida y procesada por el servidor.
- **Redirecciones (300 - 399):** Códigos de redireccionamiento devueltos cuando un nuevo recurso ha sido sustituido por el recurso solicitado.
- **Errores de los clientes (400 - 499):** Códigos de error del cliente que indican que hubo un problema con la solicitud.-
- **Errores de los servidores (500 - 599)** Códigos de error del servidor que indican que la solicitud fue aceptada, pero que un error en el servidor impidió que se cumpliera.

Dentro de cada una de estas clases, existe una variedad de códigos que pueden ser devueltos por el servidor. Cada código individual tiene un significado específico y único.

### 2.2. Códigos de respuesta del protocolo HTTP.

#### 2.2.1. Respuestas informativas (100 - 199):

Respuestas Informativas		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
100	Continue	El servidor ha recibido los headers del request y el cliente debería proceder a enviar el cuerpo de la respuesta.
101	Switching Protocol	El requester ha solicitado al servidor conmutar protocolos.
102	Processing	Usado en requests para reanudar peticiones PUT o POST abortadas.

#### 2.2.2. Respuestas satisfactorias (200 - 299):

Respuestas satisfactorias		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
200	OK	El request es correcto. Esta es la respuesta estándar para respuestas correctas.
201	Created	El request se ha completado y se ha creado un nuevo recurso.
202	Accepted	El request se ha aceptado para procesarlo, pero el proceso aún no ha terminado.
203	Non-Authoritative Information	El request se ha procesado correctamente, pero devuelve información que podría venir de otra fuente.
204	No Content	El request se ha procesado correctamente, pero no devuelve ningún contenido.
205	Reset Content	El request se ha procesado correctamente, pero no devuelve ningún contenido y se requiere que el requester recargue el contenido.

Respuestas satisfactorias		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
206	Partial Content	El servidor devuelve sólo parte del recurso debido a una limitación que ha configurado el cliente (se usa en herramientas de descarga como wget).
207	Multi-Status	El cuerpo del mensaje es XML y puede contener un número de códigos de estado diferentes dependiendo del número de sub-requests.
226	IM Used	El servidor ha cumplido una petición GET para el recurso y la respuesta es una representación del resultado de una o más manipulaciones de instancia aplicadas a la instancia actual.

### 2.2.3. Redirecciones (300 - 399):

Redirecciones		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
300	Multiple Choices	Es una lista de enlaces. El usuario puede seleccionar un enlace e ir a esa dirección. Hay un máximo de cinco direcciones.
301	Moved Permanently	La página solicitada se ha movido permanentemente a una nueva URI.
302	Found	La página solicitada se ha movido temporalmente a una nueva URI.
303	See Other	La página solicitada se puede encontrar en una URI diferente.
304	Not Modified	Indica que la página solicitada no se ha modificado desde la última petición.
305	Use Proxy	El recurso solicitado sólo está disponible a través de proxy, cuya dirección se proporciona en la respuesta. Muchos clientes HTTP como Mozilla o Internet Explorer no manejan bien estas respuestas con estos códigos de estado, sobre todo por seguridad.
307	Temporary Redirect	La página solicitada se ha movido temporalmente a otra URL.
308	Permanent Redirect	El request y futuros requests deberían repetirse usando otro URI

#### 2.2.4. Errores de los clientes (400 - 499):

Errores de los Clientes		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
400	Bad Request	El servidor no puede o no va a procesar el request por un error de sintaxis del cliente.
401	Unauthorized	Es devuelto por el servidor cuando el recurso de destino carece de credenciales de autenticación válidas.
402	Payment Required	Originalmente, este código fue creado para ser usado como parte de un sistema de dinero digital. Sin embargo, ese plan nunca se llevó a cabo y está reservado para futuro uso. Sin embargo es utilizado por diversas plataformas para indicar que una solicitud no se puede cumplir, por lo general debido a la falta de los fondos necesarios.
403	Forbidden	El request fue válido pero el servidor se niega a responder, debido a que el usuario intenta acceder a algo a que no tiene permiso para ver.
404	Not Found	El recurso del request no se ha podido encontrar.
405	Method Not Allowed	Se ha hecho un request con un recurso usando un método request no soportado por ese recurso (por ejemplo usando GET en un formulario que requiere POST).
406	Not Acceptable	El recurso solicitado solo genera contenido no aceptado de acuerdo con los headers Accept enviados en el request.
407	Proxy Authentication Required	El cliente se debe identificar primero con el proxy.
408	Request Timeout	El cliente no ha enviado un request con el tiempo necesario con el que el servidor estaba preparado para esperar.
409	Conflict	El servidor no pudo procesar la solicitud de su navegador porque hay un conflicto con el recurso correspondiente.
410	Gone	El recurso solicitado no está disponible ni lo estará en el futuro.
411	Length Required	El request no especificó la longitud del contenido, la cual es requerida por el recurso solicitado.
412	Precondition Failed	El servidor no cumple una de las condiciones que el requester añade en el request.
413	Request Entity Too Large	El request es más largo que el que está dispuesto a aceptar el servidor.
414	Request-URI Too Long	El URI es muy largo para que el servidor lo procese.
415	Unsupported Media Type	La entidad request tiene un media type que el servidor o recurso no soportan.

Errores de los Clientes		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
416	Requested Range Not Satisfiable	La solicitud fue por una porción de un recurso que el servidor no puede devolver.
417	Expectation Failed	El servidor no puede cumplir los requisitos especificados en el campo de cabecera de la solicitud.
418	I'm a teapot	Fue parte de un April Fool's day, y no se espera que se implemente en servidores HTTP. La RFC especifica que este código debería ser devuelto por teteras para servir té.
421	Misdirected Request	La petición fue dirigida a un servidor que no es capaz de producir una respuesta. Esto puede ser enviado por un servidor que no está configurado para producir respuestas por la combinación del esquema y la autoridad que están incluidos en la URI solicitada.
422	Unprocessable Entity	La petición estaba bien formada pero no se pudo seguir debido a errores de semántica.
423	Locked	El recurso que está siendo accedido está bloqueado.
424	Failed Dependency	La petición falló debido a una falla de una petición previa.
426	Upgrade Required	El servidor se rehúsa a aplicar la solicitud usando el protocolo actual pero puede estar dispuesto a hacerlo después que el cliente se actualice a un protocolo diferente. El servidor envía un encabezado Upgrade en una respuesta para indicar los protocolos requeridos.
428	Precondition Required	El servidor origen requiere que la solicitud sea condicional.
429	Too Many Requests	El usuario ha enviado demasiadas solicitudes en un periodo de tiempo dado.
431	Request Header Fields Too Large	El servidor no está dispuesto a procesar la solicitud porque los campos de encabezado son demasiado largos.
451	Unavailable For Legal Reasons	El usuario solicita un recurso ilegal, como alguna página web censurada por algún gobierno.

### 2.2.5. Errores de los servidores (500 - 599):

Errores del Servidor		
<i>Código</i>	<i>Nombre</i>	<i>Descripción</i>
500	Internal Server Error	Error genérico, cuando se ha dado una condición no esperada y no se puede concretar el mensaje.
501	Not Implemented	El servidor o no reconoce el método del request o carece de la capacidad para completarlo.
502	Bad Gateway	El server actuaba como puerta de entrada o proxy y recibió una respuesta inválida del servidor upstream.
503	Service Unavailable	El servidor está actualmente no disponible, ya sea por mantenimiento o por sobrecarga.
504	Gateway Timeout	El servidor estaba actuando como puerta de entrada o proxy y no recibió una respuesta oportuna por parte del servidor upstream.
505	HTTP Version Not Supported	El servidor no soporta la versión del protocolo HTTP usada en el request.
506	Variant Also Negotiates	El servidor tiene un error de configuración interna(negociación de contenido transparente para la petición resulta en una referencia circular).
507	Insufficient Storage	El servidor tiene un error de configuración interna(a variable de recurso escogida está configurada para acoplar la negociación de contenido transparente misma, y no es por lo tanto un punto final adecuado para el proceso de negociación).
508	Loop Detected	El servidor detectó un ciclo infinito mientras procesaba la solicitud.
510	Not Extended	Extensiones adicionales para la solicitud son requeridas para que el servidor las cumpla.
511	Network Authentication Required	El cliente necesita autenticarse para poder acceder a la red.



### 3. Conclusión

A partir del uso de los códigos de estado de respuesta que provee HTTP, se puede tener un conocimiento amplio acerca de como se comporta una aplicación Web cuando recibe peticiones por parte de uno o más clientes, de tal manera que se puede identificar si la aplicación esta brindando respuestas adecuadas, y en caso contrario saber si se trata de un error por parte del cliente o del servidor y conocer más a detalle que está causando dicho error durante el proceso de intercambio de información entre el servidor y el cliente para su posterior corrección.

Para lo último mencionado los códigos del 400-499 (Errores del cliente) y del 500-599 (Errores del servidor) son de bastante utilidad para identificar dichos errores de manera específica, así como también los más comunes de aparecer, pues incluso se pueden encontrar en el día a día al navegar por internet, existe la posibilidad de que cuando se acceda a una página Web haya algún error que nos muestre algún código perteneciente a los rangos anteriormente mencionados.

*Mauro Sampayo Hernández*

## 4. Referencias Bibliográficas

### Referencias

- [1] *Códigos de estado de respuesta HTTP* MDN Web Docs [accessed 2021 Oct 27] <https://developer.mozilla.org/es/docs/Web/HTTP/Status>
- [2] Diego Lázaro. *Códigos de estado HTTP* **2018** [accessed 2021 Oct 27] <https://diego.com.es/codigos-de-estado-http>
- [3] Jon Penland. *Una guía completa y una lista de códigos de estado HTTP* KINSTA, **2021** [accessed 2021 Oct 27] <https://kinsta.com/es/blog/codigos-de-estado-de-http/>