Assignment 2 – HTTP Status Numbers

Sasanka Vepakomma

Ira A. Fulton Schools of Engineering, Arizona State University

IFT 598: Middleware Programming & Database Security

Prof. Dinesh Sthapit

Due: Sept 3, 2022

HTTP Response status codes

1xx Category (Information Responses):

This is called informational category. This category is not used in any websites and will not be encountered by users or need to be programmed while creating any REST APIs.

2xx Category (Successful Responses):

This category is called successful responses. The HTTP response codes which are of 2xx format are shown when a request is successfully received by the server and it responds with the expected outcome. The following are some of the important 2xx responses:

200 OK:

This indicates the response is received successfully. Success can mean different things based on the request made. For a GET request, success means the response getting captured successfully in the response message. For a HEAD request, the correct representation of headers gets included in the response message. For a PUT or POST request, certain record(s) in the server get updated and the result gets captured in the message body. A TRACE request gives back the message as it is received by the server in the response message body.

201 Created:

This status code in response indicates a successful creation of a resource. This response is seen in cases like creation of users, shopping cart items, accounts, etc. indicating that the desired resource has been created. A status 200 OK response also works in this case. However, a 201 is used to give more specific information to the browser regarding the response in server.

204 No Content:

This status code represents that the request has been received and executed successfully, however, there is nothing to be returned from the server. For example, when a delete request has been made, oftentimes no response is programmed to be sent back as it’s a delete action.

3xx Category (Redirection Messages):

This category of responses are seen when a website redirects the user to a different website from the current requested one. Based on the level of redirection, different response codes are sent by the server. The most common one among them is the 304.

304 Not Modified:

Oftentimes users may request large information which gets stored in the browser cache. This response is sent by the server periodically to indicate if any changes have been made with the APIs, and if a corresponding change in the browser cache may be needed. When this response is received, it means that no changes have been made with API and no cache flush is required.

4xx Category (Client Error):

The 400 level errors are client-side errors. The API uses these response codes to convey to the user that the request sent by them has errors and will need to be fixed.

400 Bad Request:

This error is sent by the server when it fails to process the user’s request due to erroneous request. This can include requests with missing fields, wrong API name, invalid syntax, etc.

401 Unauthorized:

This response code is sent by the server when users make requests without passing their authorization details. The request may have an authentication cookie, an API key, etc missing, and this response is sent back by the server. As a result, the requested data is prevented from user access.

403 Forbidden:

This response is sent by the server when the user is authenticated, but is not allowed to access the requested content.

404 Not Found:

This error is seen when user tries to access a resource that is non-existent. For example, when a user tries to access a shopping cart API while a website offers locations API, the requesting users will receive this error code.

5xx Category (Server Error):

Essentially 5xx errors are seen when there is a malfunction from the server side. A 5xx error indicates that the request sent by the user is flawless, and there is some issue from the server side, which needs to be fixed.

500 Internal Server Error:

This is the most common error seen when a server is broken or when a database is down.

HTTP Response headers

A response header is an HTTP header that can be used in an HTTP response and that doesn't relate to the content of the message. Response headers, like Age, Location or Server are used to give a more detailed context of the response. An example of a HTTP response that depicts a few headers after a GET request:

200 OK

Access-Control-Allow-Origin: \*

Connection: Keep-Alive

Content-Encoding: gzip

Content-Type: text/html; charset=utf-8

Date: Mon, 18 Jul 2016 16:06:00 GMT

Etag: "c561c68d0ba92bbeb8b0f612a9199f722e3a621a"

Keep-Alive: timeout=5, max=997

Last-Modified: Mon, 18 Jul 2016 02:36:04 GMT

Server: Apache

Set-Cookie: mykey=myvalue; expires=Mon, 17-Jul-2017 16:06:00 GMT; Max-Age=31449600; Path=/; secure

Transfer-Encoding: chunked

Vary: Cookie, Accept-Encoding

X-Backend-Server: developer2.webapp.scl3.mozilla.com

X-Cache-Info: not cacheable; meta data too large

X-kuma-revision: 1085259

x-frame-options: DENY