

WEB SISTEMAK

2020/2021 IKASTURTEA

HTTP - HyperText Transfer Protocol

Eskaera eta erantzuna. Adibidea Python erabiliz: *requests* liburutegia

Errore kodeak: 400 eta 404.

Berhelbideraketak (3xx erantzun kodeak)

Web orri baten eskaeratik eratorritako trafikoa. Nabigatzailearen garapen tresnak.

2021-02-05 M



Web Sistemak by [Oskar Casquero](#) & [María Luz Álvarez](#) is licensed under a [Creative Commons Reconocimiento 4.0 Internacional License](#).

HTTP-REN FUNTZIONAMENDUA: BEZEROAREN ESKAERA

HTTP eskaeraren sintaxia

Metodoa URIa HTTP/1.1
Goiburuak
CRLF
Mezuaren gorputza

Adibidearen HTTP eskaera

GET /baliabidea HTTP/1.1
Host: ws2021.com:8080
Accept: text/html
Accept-Encoding: gzip,identity;q=0.5
Accept-Language: en-US,es-ES;q=0.8
User-Agent: Mozilla Windows Desktop

Metodoa: GET

Metodoak baliabidearen gainean zein CRUD (Create, Read, Update and Delete) ekintza mota burutu nahi den adierazten du. Kasu honetan, GET → irakurketa.

URIa: /baliabidea

Baliabidearen identifikazioa URI osoarekin edo URI erlatiboarekin egin daiteke.

GET http://ws2021.com:8080/baliabidea HTTP/1.1

GET /baliabidea HTTP/1.1

Host: ws2021.com:8080

Goiburuak: bezeroaren ezaugarriak eta erantzunarekiko lehenetasunak adierazten dituzte.

Accept: eduki bezela HTML onartzen dela adierazten da.

Accept-Encoding: nabigatzailearentzako eduki konprimatuak (gzip formatuan) lehenetasuna dauka, baina konprimatu gabeko edukia (identity) ere onartzen du.

Accept-Language: nabigatzaileak bere hizkuntz nagusia *en* dela adierazten du, bigarren aukera *es* delarik.

User-Agent: nabigatzaileak bere burua mahaigaineko Windows batean dagoen Mozilla bezela aurkezten du.

Mezuaren gorputza: kasu honetan hutsik dago.

HTTP-REN FUNTZIONAMENDUA:

ZERBITZARIAREN ERANTZUNA

HTTP erantzunaren sintaxia

HTTP/1.1 Status Deskribapena

Goiburuak

CRLF

Mezuaren gorputza (zortzikoteetan)

Adibidearen HTTP erantzuna

HTTP/1.1 200 OK

Date: Thu, 20 Nov 2015 20:25:52 GMT

Last-Modified: Tue, 17 Sep 2015 13:00:02 GMT

ETag: "1a968-3ec-4e693e61bb8b6"

Content-Length: 76

Content-Type: text/html; charset=ISO-8859-1

```
<html><head><title>index.html</title></head>
```

```
<body>Hello World!</body></html>
```

Status: 200

Eskaera ulertzeko eta betetzeko ahalegina deskribatzen duen kodea.

[200 kodeak](#) eskaera ondo osoturik dagoela eta zuzen prozesatu dela adierazten du.

Programei zuzendua.

Deskribapena: OK

Status-ari loturiko testu deskribapena.

Gizakiei zuzendua.

Goiburuak: erantzunaren alderdi ezberdinak bereizten ditu.

Content-Length: zerbitzariak mezuaren gorputzeko luzeera (zortzikote kopurua) adierazten du.

Content-Type: zerbitzariak edukia HTML motakoa dela eta bere zortzikoteak *latin-1*-en (ISO-8859-1) kodifikatuta daudela adierazten du.

Mezuaren gorputza: edukia; kasu honetan, HTML dokumentua (web orria)

ADIBIDEA PYTHON ERABILIZ: *REQUESTS* LIBURUTEGIA

- Python-en *requests* liburutegia erabiliz, ondorengo baliabidea eskatu:
<http://www.httpwatch.com/httpgallery/chunked/chunkedimage.aspx>

```
import requests

# Eskerak 4 atal ditu: metodoa, uria, goiburuak eta edukia
metodoa = 'GET'
uria = "http://www.httpwatch.com/httpgallery/chunked/chunkedimage.aspx"
goiburuak = {'Host': 'www.httpwatch.com'}
edukia = ''
erantzuna = requests.request(metodoa, uria, headers=goiburuak, data=edukia)

# Erantzunak 4 atal ditu baita: kodea, deskribapena, goiburuak eta edukia
kodea = erantzuna.status_code
deskribapena = erantzuna.reason
print(str(kodea) + " " + deskribapena)
for goiburua in erantzuna.headers:
    print(goiburua + ": " + erantzuna.headers[goiburua])
edukia = erantzuna.content
print(edukia)

fitxategia = open("irudia.jpg", 'wb')
fitxategia.write(edukia)
fitxategia.close()
```

HTTP-REN FUNTZIONAMENDUA: ERROREAK

- **Zer gertatzen da eskaera bete ezin daitekeenean?**

- 4xx eta 5xx erantzun kodeak**

- Bezeroak edo zerbitzariak eraginda, eskaera bat bete ezin denean, erantzunean ondorengo Status kodeak itzultzen dira.

Bezeroak eragindako erroreak:

4xx ([RFC 2616, 10.4 atala](#))

- 400 Bad Request ←
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found ←
- ...

Zerbitzariak eragindako erroreak:

5xx ([RFC 2616, 10.5 atala](#)):

- 500 Internal Server Error
- 503 Service Unavailable
- ...

HTTP-REN FUNTZIONAMENDUA: 400 ERROREA

- 400 Bad Request

- Zerbitzariak ez du eskaera ulertzen bere semantika/sintaxia okerrak direlako.
- Demagun */html/main page.html* baliabidea eskatzen dela.

Eskaera baten adibidea

```
GET /html/main page.html HTTP/1.1
Host: ws2021.com
Accept: text/html
Accept-Encoding: gzip,identity;q=0.5
Accept-Language: en-US,es-ES;q=0.8
User-Agent: Mozilla Windows Desktop
```

Erantzun baten adibidea

```
HTTP/1.1 400 Bad Request
Date: Wed, 25 Nov 2015 08:07:43 GMT
Content-Length: 138
Content-Type: text/html; charset=UTF-8

<html><head><title>Error 400 (Bad
Request)</title></head><body><p>Your
client has issued a malformed or illegal
request.</p></body></html>
```

Zerbitzariak eskaera irakurtzean ulertzen duena:

- Metodoa: GET
- URI: /html/main
- Protokoloaren bertsioa: page.html → **SEMANTIKA ERROREA** (kate hau HTTP/1.1 izan behar da)
- Kate gehigarria: HTTP/1.1 → **SINTAXI ERROREA** (lehenengo lerroak 3 osagai izan behar ditu)

Konponbidea: URI-an zuriunea kodifikatzea: /html/main%20page.html

ADIBIDEA: 400 ERROREA (SEMANTIKA ERROREA)

- Aurreko adibidearen kodean, Host goiburuaren izenaren amaieran zuriune bat sartu

```
import requests

# Eskaerak 4 atal ditu: metodoa, uria, goiburuak eta edukia
metodoa = 'GET'
uria = "http://www.httpwatch.com/httpgallery/chunked/chunkedimage.aspx"
goiburuak = {'Host ': 'www.httpwatch.com'}
edukia = ''
erantzuna = requests.request(metodoa, uria, headers=goiburuak, data=edukia)

# Erantzunak 4 atal ditu baita: kodea, deskribapena, goiburuak eta edukia
kodea = erantzuna.status_code
deskribapena = erantzuna.reason
print(str(kodea) + " " + deskribapena)
for goiburua in erantzuna.headers:
    print(goiburua + ": " + erantzuna.headers[goiburua])
edukia = erantzuna.content
print(edukia)

fitxategia = open("irudia.jpg", 'wb')
fitxategia.write(edukia)
fitxategia.close()
```

HTTP-REN FUNTZIONAMENDUA: 404 ERROREA

- 404 Not Found

- Eskaera ondo formateatuta dago, baina zerbitzariak ez du URI-a aurkitzen.
- Demagun *sw2016.com* zerbitzariari */login/index.php* baliabidea eskatzen zaiola, zerbitzari horretan existitzen ez dena.

Eskaera baten adibidea

```
GET /login/index.php HTTP/1.1
Host: ws2021.com
Accept: text/html
Accept-Encoding: gzip,identity;q=0.5
Accept-Language: en-US,es-ES;q=0.8
User-Agent: Mozilla Windows Escritorio
```

Erantzun baten adibidea

```
HTTP/1.1 404 Not Found
Date: Wed, 25 Nov 2015 08:07:43 GMT
Content-Length: 134
Content-Type: text/html; charset=UTF-8

<html><head><title>Error 404 (Not
Found)</title></head><body><p>The
requested resource was not found in this
server.</p></body></html>
```


ADIBIDEA: 404 ERROREA


– Aurreko adibidearen kodean, **baliabidearen** izena aldatu: `/httpgallery/chunked/chunkedtext.aspx`

```
import requests

# Eskaerak 4 atal ditu: metodoa, uria, goiburuak eta edukia
metodoa = 'GET'
uria = "http://www.httpwatch.com/httpgallery/chunked/chunkedtext.aspx"
goiburuak = {'Host': 'www.httpwatch.com'}
edukia = ''
erantzuna = requests.request(metodoa, uria, headers=goiburuak, data=edukia)

# Erantzunak 4 atal ditu baita: kodea, deskribapena, goiburuak eta edukia
kodea = erantzuna.status_code
deskribapena = erantzuna.reason
print(str(kodea) + " " + deskribapena)
for goiburua in erantzuna.headers:
    print(goiburua + ": " + erantzuna.headers[goiburua])
edukia = erantzuna.content
print(edukia)

fitxategia = open("irudia.jpg", 'wb')
fitxategia.write(edukia)
fitxategia.close()
```



HTTP-REN FUNTZIONAMENDUA: BERHELBIDERAKETAK

- **Zer dira berhelbideraketak?**

- Batzutan, baliabide baten URI-a aldatu edo egokitu daiteke. Adibidez, 2017-18 kurtsoan, <http://egela.ehu.eus/> URI-a <http://egela2017-18.ehu.eus/> URI-ra zuzenduta zegoen; aldiz, 2018-19 kurtsoan <http://egela2018-19.ehu.eus/> URI-ra zuzenduta zegoen.
- Kasu bietan, erabiltzaileak ez zuen benetako URI-a ezagutu behar, <http://egela.ehu.eus/> eskatzerakoan nabigatzaileak erabiltzailea automatikoki zegoen URI-ra berhelbideratzen baitzuen.

- **Zelan egiten dira berhelbideraketak?**

- HTTP protokoloak beste URI batera berhelbideraketa egiteko modua eskeintzen du. Horretarako,
 - 301, 302 edo 303 erantzun kodeak
 - eta “Location” goiburua erabiltzen dira.

HTTP-REN FUNTZIONAMENDUA: BERBIDERAKETAK

- Demagun Madrilen kokatutako bezero batek ondorengo URI-a duen baliabidea eskatzen duela: <http://www.google.com/>
- HTTP eskaera hori erantzuten duen zerbitzariak paketearen jatorrizko IP helbidea Espainiakoa dela antzematen du: bezeroari <http://www.google.es/> URI-ra berhelbideraketa egiteko erantzuna bidaltzen dio.
- Bezeroak, erantzunean 302 kodea detektatzean, “Location” goiburuaren balioa atera eta URI berrira beste eskaera bat egiten du. Nabigatzailean prozesu hau erabiltzailearentzako modu gardenean gertatzen da.

Eskaera baten adibidea

```
GET / HTTP/1.1
Host: www.google.com
Accept: text/html
Accept-Encoding: identity
Accept-Language: en-US,es-ES;q=0.8
User-Agent: Mozilla Windows Desktop
```

Erantzun baten adibidea

```
HTTP/1.1 302 Found
Content-Length: 137
Content-Type: text/html; charset=UTF-8
Location: http://www.google.es/

<html><head><title>Redirection
302</title></head><body><a
href="http://www.google.es/">Redirect to
http://www.google.es/</a></body></html>
```

ADIBIDEA NABIGATZAILEAREN GARAPEN TRESNAK ERABILIZ

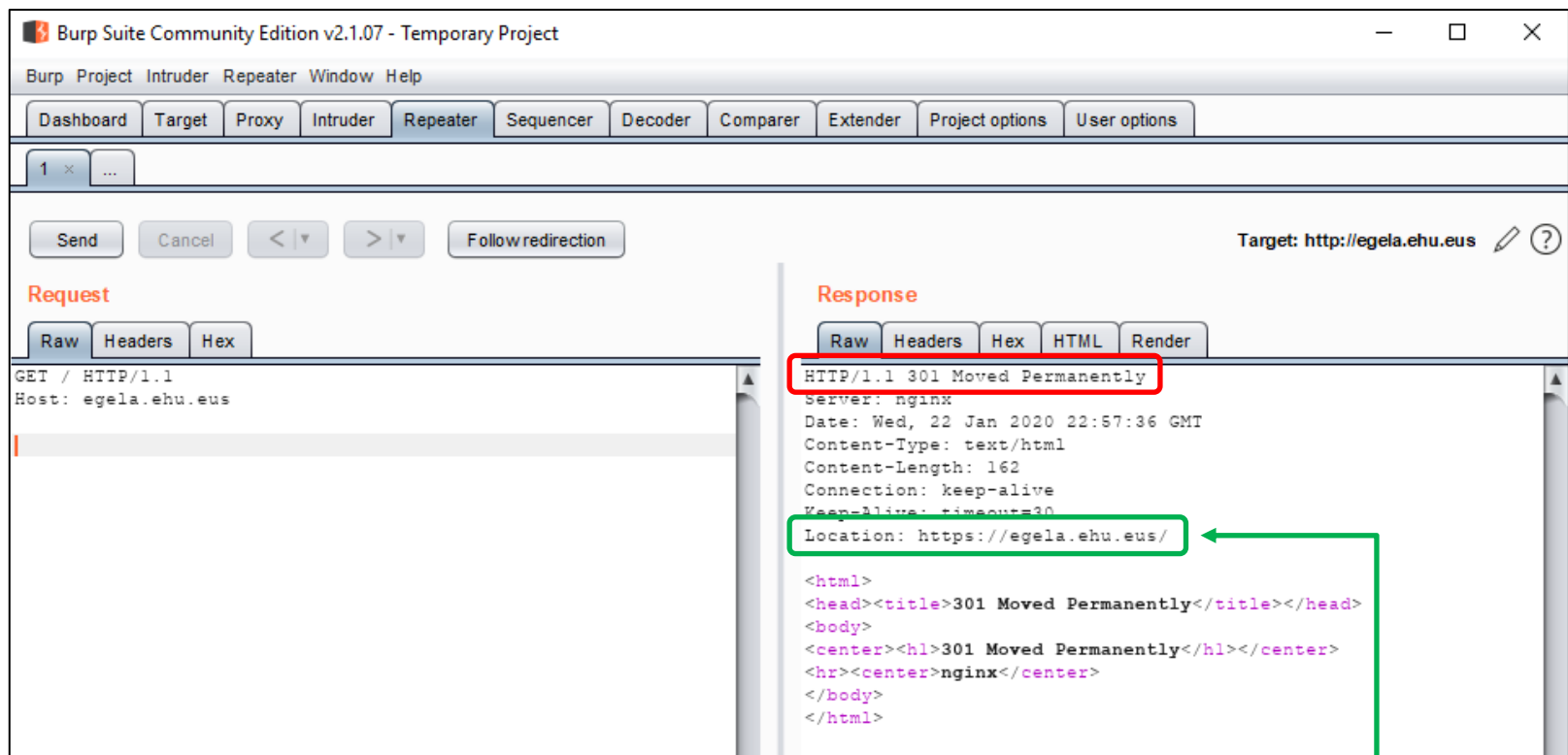
- Nabigatzailean erlaitz berri bat zabaldu, bertan garatzailearen tresnak erabiliz (F12)
<http://egela.ehu.eus/> baliabidea eskatu

The screenshot shows the eGela 2019-20 UPV/EHU website in a browser. The Chrome DevTools Network tab is open, displaying a list of resources loaded by the page. The resources are organized into a table with columns: Name, Status, Type, Initiator, Size, Time, and Waterfall. The first two resources, 'egela.ehu.eus' and 'egela.ehu.eus', are highlighted with a red box. The table shows various types of resources including HTML documents, CSS stylesheets, JavaScript files, images, and fonts. The status column indicates that all resources were loaded successfully (200). The time column shows the time taken to load each resource, and the waterfall column provides a visual representation of the loading sequence.

Name	Status	Type	Initiator	Size	Time	Waterfall
egela.ehu.eus	301	text/html	Other	214 B	82 ms	
egela.ehu.eus	200	document	egela.ehu.eus/	12.9 KB	329 ms	
yui_combo.php?rollup/3.17.2/yui-moodiesimple-min.css	200	stylesheet	(index)	1.5 KB	29 ms	
all	200	stylesheet	(index)	132 KB	360 ms	
yui_combo.php?rollup/3.17.2/yui-moodiesimple-min.js	200	script	(index)	82.9 KB	363 ms	
jquery-3.2.1.min.js	200	script	(index)	30.0 KB	123 ms	
javascript-static.js	200	script	(index)	7.1 KB	89 ms	
ResizeSensor.js	200	script	(index)	8.8 KB	87 ms	
ElementQueries.js	200	script	(index)	19.9 KB	113 ms	
js?id=UA-98997857-3	200	script	(index)	27.6 KB	127 ms	
logoECAMPUS.png	200	png	(index)	1.8 KB	28 ms	
require.min.js	200	script	(index)	6.9 KB	28 ms	
scrolltop.js	200	script	(index)	768 B	27 ms	
tooltipfix.js	200	script	(index)	658 B	29 ms	
header_egela.svg	200	svg+xml	(index)	478 KB	557 ms	
egela1819.png	200	png	(index)	157 KB	845 ms	
GAV.png	200	png	(index)	219 KB	258 ms	
ceu.png	200	png	(index)	237 KB	732 ms	
yui_combo.php?3.17.2/cssbutton/cssbutton-min.css	200	stylesheet	yui_combo.php?rollup/3.1...	1.6 KB	27 ms	
fontawesome-webfont.woff2?v=4.7.0	200	font	(index)	75.8 KB	105 ms	
first.js	200	script	require.min.js	260 KB	357 ms	
data:image/svg+xml=...	200	svg+xml	jquery-3.2.1.min.js	(memory cache)	0 ms	
data:image/svg+xml=...	200	svg+xml	jquery-3.2.1.min.js	(memory cache)	0 ms	
analytics.js	200	script	js?id=UA-98997857-3	17.8 KB	80 ms	
EgelaPortfolio.png	200	png	jquery-3.2.1.min.js	82.2 KB	184 ms	
collect?v=1&aip=1&dt=2021447936&t=pageview&s=1&d...	302	text/html	analytics.js	301 B	46 ms	
collect?v=1&aip=1&dt=2021447936&t=pageview&s=1&d...	200	gif	collect	407 B	123 ms	
blackboard-collaborate.jpg	200	jpeg	jquery-3.2.1.min.js	35.6 KB	72 ms	
yui_combo.php?m/1579685327/core/event/event-min.js...	200	script	yui_combo.php?rollup/3.1...	6.7 KB	27 ms	
yui_combo.php?m/1579685327/filter_mathjaxloader/loader...	200	script	yui_combo.php?rollup/3.1...	1.1 KB	26 ms	
event.js	200	script	require.min.js	260 KB	247 ms	
jquery-3.2.1.min.js	200	script	require.min.js	29.9 KB	52 ms	
yui_combo.php?3.17.2/event-mousewheel/event-mousew...	200	script	yui_combo.php?rollup/3.1...	5.2 KB	27 ms	
service.php?sesskey=nDpEUP36n&info=c_fetch_notifica...	200	xhr	jquery-3.2.1.min.js	577 B	47 ms	

ADIBIDEA BURP ERABILIZ

- Burp-en *Repeater* erlaitza erabiliz, ondorengo baliabidea eskatu:
<http://egela.ehu.eus/>



- Zein da eGelako orrialde nagusia lortzeko hurrengo pausua?

ADIBIDEA PYTHON ERABILIZ

– Python-en *requests* liburutegia erabiliz, aurreko adibidea errepikatu.

```
import requests

metodoa = 'GET'
uria = "http://egela.ehu.eus/"
goiburuak = {'Host': 'egela.ehu.eus'}
edukia = ''
erantzuna = requests.request(metodoa, uria, headers=goiburuak, data=edukia, allow_redirects=False)

kodea = erantzuna.status_code
deskribapena = erantzuna.reason
print(str(kodea) + " " + deskribapena)
for goiburua in erantzuna.headers:
    print(goiburua + ": " + erantzuna.headers[goiburua])
edukia = erantzuna.content
print(edukia)

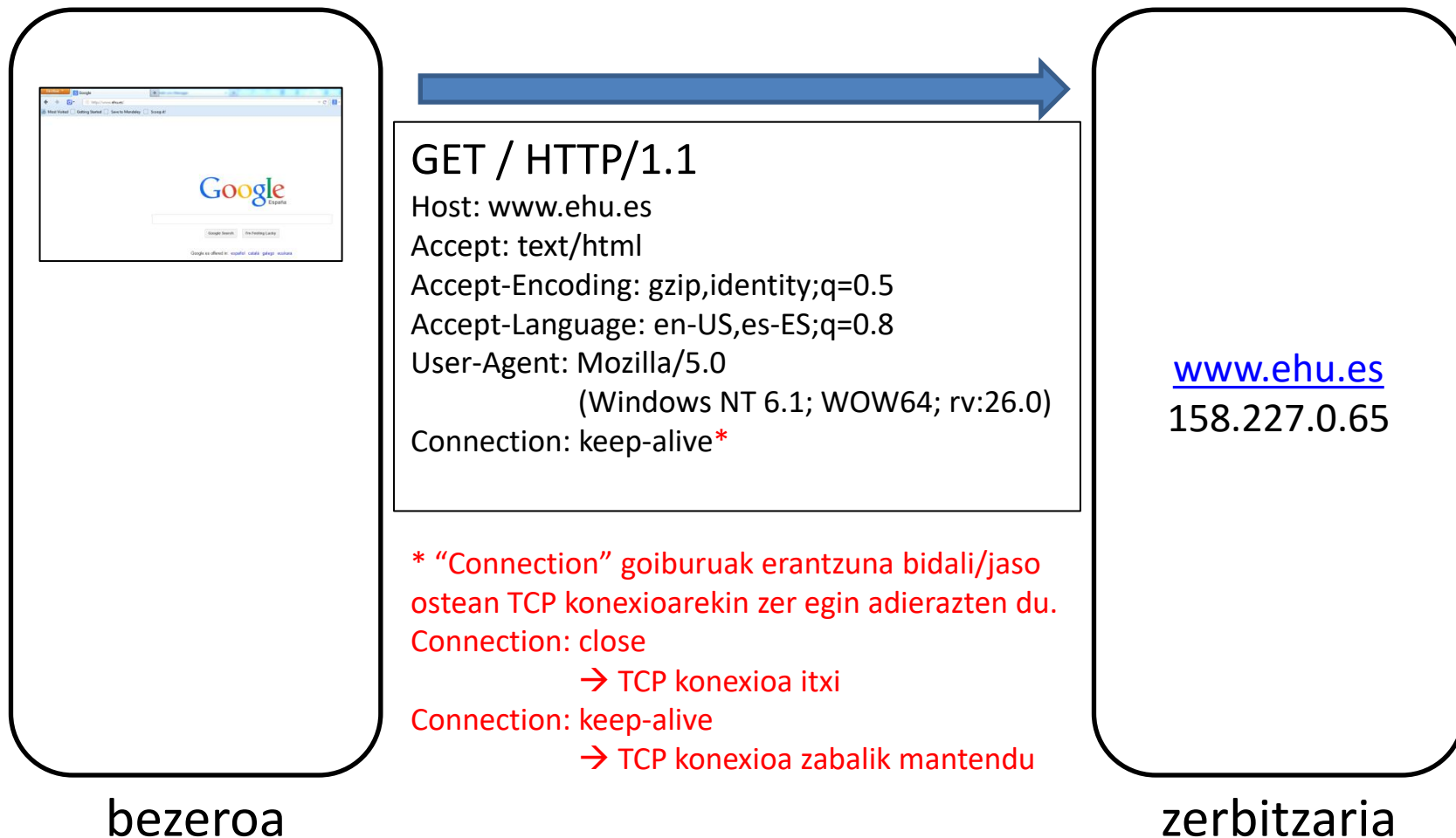
metodoa = 'GET'
uria = erantzuna.headers['Location']
goiburuak = {'Host': uria.split('/')[2]}
edukia = ''
erantzuna = requests.request(metodoa, uria, headers=goiburuak, data=edukia, allow_redirects=False)

kodea = erantzuna.status_code
deskribapena = erantzuna.reason
print(str(kodea) + " " + deskribapena)
for goiburua in erantzuna.headers:
    print(goiburua + ": " + erantzuna.headers[goiburua])
edukia = erantzuna.content
print(edukia)
```

URI berrian host-a aldatu daitekela kontuan hartu!!!

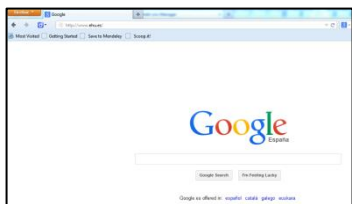
HTTP-REN FUNTZIONAMENDUA:

WEB ORRIALDE BATEN KARGA



HTTP-REN FUNTZIONAMENDUA:

WEB ORRIALDE BATEN KARGA



bezeroa

* “Keep-Alive” goiburuak TCP konexioa mantentzeko baldintzak zehazten ditu.

timeout=2 → TCP konexio bat HTTP trafiko gabe mantendu daitekeen denbora (s-tan) adierazten du.

max=500 → TCP konexio baten barnean bidali daitekeen HTTP eskaera kopuru maximoa.



HTTP/1.1 200 OK

Date: Thu, 20 Nov 2015 20:25:52 GMT

Last-Modified: Tue, 17 Sep 2015 13:00:02 GMT

ETag: "1a968-3ec-4e693e61bb8b6"

Content-Length: 1004

Content-Type: text/html; charset=ISO-8859-1

Keep-Alive: timeout=2, max=500*

WEB ORRIA (HTML DOKUMENTUA)

www.ehu.es

158.227.0.65

zerbitzaria

HTTP-REN FUNTZIONAMENDUA:

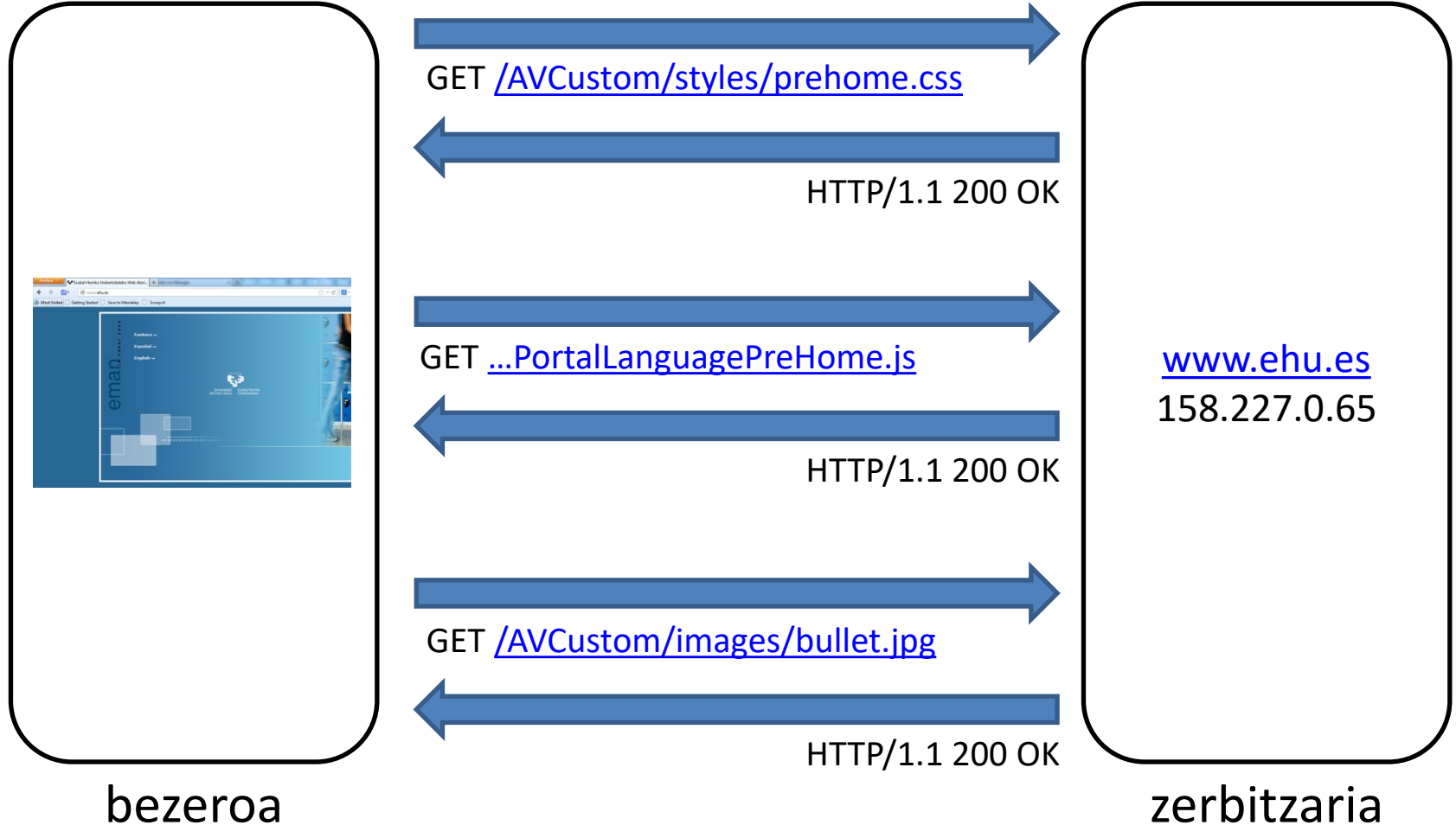
WEB ORRIALDE BATEN KARGA

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
Transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="eu">
  <head>
    <title>Euskal Herriko Unibertsitateko Web Ataria/Portal web de la Universidad del País Vasco</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
    <link href="/AVCustom/styles/prehome.css" rel="stylesheet" type="text/css"/>
    <script type="text/javascript" src="/AVCustom/r01gLangSelectorVA/scripts/PortalLanguagePreHome.js"/>
  </head>
  <body>
    <script type="text/javascript"> testPreHome("p200","home","home","shenhm"); </script>
    <div id=container>
      <ul>
        <li><a href="/p200-home/eu/">Euskara </a></li>
        <li><a href="/p200-home/es/">Español </a></li>
        <li><a href="/p200-shenhm/en">English </a></li>
      </ul>
    </div>
  </body>
</html>
```

WEB ORRIA
(HTML DOKUMENTUA)

HTTP-REN FUNTZIONAMENDUA:

WEB ORRIALDE BATEN KARGA



HTTP-REN FUNTZIONAMENDUA:

TCP KONEXIOAREN KUDEAKETA

- Aurreko adibideko web orria kargatzeko 4 HTTP transferentzia egin behar dira:
 - HTML orria
 - irudia
 - CSS estilo orria
 - JavaScript kodea
- HTTP eskaera bati lotutako baliabideen deskarga ondorengo moduetan egin daiteke:
 - TCP konexio berdina erabilita, konexio iraunkorra ere deitua (eraginkorra)
 - TCP konexio bereizietan (ez eraginkorra)
 - Modu mistoan.
- Azpiko TCP konexioak kudeatzeko, HTTP protokoloak bi goiburu ditu:
 - Eskaeran: “Connection”
 - Erantzunean: “Keep-Alive”

HTTP-REN FUNTZIONAMENDUA:

TCP KONEXIOAREN KUDEAKETA

- Eraginkortasuna TCP konexio ezarpen (SYN, SYN-ACK, ACK) kopuruaren arabera neurtzen da.
- HTTP/1.1-ek TCP konexio iraunkorrak erabiltzen dituen arren, praktikan, nabigatzaile batek jatorri bakoitzeko batazbesteko 4-8 konexio zabaltzen ditu.
- Web orri bat jatorri askotako edukiak hartuta osotzen dela kontutan izanda, ohiko web orri batek 30 TCP konexio baino gehiago ezar ditzake, honek suposatzen duen gainkargarekin.
- Jatorri bakoitzeko TCP konexio bakarra erabili behar da.

UPV/EHU-KO WEB ORRIALDE NAGUSIAREN DESKARGAREN AZTERKETA

– Nabigatzailean erlaitz berri bat zabaldu eta bertan garatzailearen tresnak abiarazi (F12)

The image shows a web browser window displaying the UPV/EHU website. The browser's developer tools are open, showing the Network tab. A red box highlights a group of social media icons (Twitter, Facebook, Instagram, LinkedIn, YouTube, Vimeo) in the website's footer, labeled 'Jarraitu!'. Another red box highlights a group of social media icons (Twitter, Facebook, Instagram, LinkedIn, YouTube, Vimeo) in the browser's network tab, which are being loaded by the website. The network tab shows a list of resources, including social media icons, images, and scripts. The status of the resources is '200', indicating they were loaded successfully. The size of the resources is shown in KB, and the time taken to load them is shown in ms. The waterfall view on the right shows the sequence of requests and responses over time.