

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

Actividad 03.

Desarrollo de Aplicaciones en Red

Profesora: Belén Bermejo

Universidad Internacional de la Rioja

21 de enero de 2021

Interacción con un Servidor Web

José Antonio Gil Valencia
08862804F

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

Índice

<u>1-Definición de la Actividad.</u>	3
<u>1.1.-Descripción y Pautas</u>	3
<u>1.2.-Descripción y Pautas</u>	3
<u>2-Desarrollo de la Actividad.</u>	4
<u>2.1.-Acceso al Servidor</u>	4
<u>2.2.-Descarga de ficheros</u>	4
<u>2.3.-Clases y Métodos de <i>JSoup</i></u>	6
<u>2.4.-Presentación de los <i>posts</i></u>	10
<u>2.3.-Conclusión</u>	13
<u>3-Referencias. Bibliografía. Webgrafía.</u>	15
<u>3.1.-Maven y Scrapping en Eclipse</u>	15
<u>3.2.-JSoup</u>	15
<u>3.3.-Java</u>	16
<u>3.4.-Salida a HTML desde Java</u>	16
<u>3.5.-AJAX</u>	16

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

1.-Definición de la Actividad.

Trabajo: Interacción con un servidor

1.1.-Descripción y pautas de elaboración.

Para realizar la siguiente actividad debemos acceder al aula virtual y descargar de manera ordenada (mediante acceso SOAP, REST y JavaScript) todas las conversaciones de todos los foros.

Estas conversaciones deben aparecer indexadas en una página web única y donde cada conversación aparezca ordenada por intervención en el tiempo.

1.2.-Entrega.

Debemos de entregar un resumen narrado del ejercicio, escrito usando tecnologías web.

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

2.-Desarrollo de la Actividad.

2.1.-Acceso al Servidor (Estructura).

La forma de llevar a cabo la actividad ha sido desarrollando un programa en Java que permita acceder a un servidor determinado y, a través de una serie de bucles, poder ir extrayendo la información de un foro determinado (en abierto) en formato HTML.

Para ello, he utilizado la herramienta de Java *JSoup*, ya que posee una amplia librería para poder acceder a un servidor y poder ir extrayendo su contenido a partir de las etiquetas del propio código HTML de la página web.

Una vez obtenido el documento HTML con todo el contenido requerido del foro, la presentación del contenido se ha presentado en un proyecto web (HTML) que, al interactuar el usuario mediante un botón, aparece el contenido del HTML del foro con todas las conversaciones indexadas de todas las páginas. Para llevar a cabo esta interacción, utilicé un código Javascript para simular la interacción del usuario con dicho documento, empleando AJAX para no tener que cambiar de página al hacer click sobre el botón.

A continuación, se presenta la descripción de cada apartado de la actividad, teniendo en cuenta que dicha descripción también se recoge en un enlace interno dentro de la propia página web presentada en la actividad (https://gilvalencia.github.io/DAR_act03/).

2.2.-Descarga de Ficheros (Java y JSoup).

En los documentos anexos a esta memoria, se encuentra el proyecto de Java con la implementación requerida para hacer el método de *scrapping* sobre la URL <https://www.verema.com/foros/hosteleria/temas>.

Este trabajo se ha realizado con la librería *JSoup* de Java. En particular con sus clases Document, Elements y Element, y sus diversos métodos que, en función de la

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

etiqueta o información a recuperar de la web, se utilizaban unos u otros (`select()`, `absUrl()`, `getElementByTag()`, `getElementByClass()`, entre otros).

Debido a la estructura de la web seleccionada, la extracción se ha tenido que realizar por partes:

- El método `getStatusConnectionCode()` previamente devuelve el código de estado Http en la que se encuentra la web. Si se encuentra en cualquier otro estado que no sea el 200 (OK), salta una excepción.

```

33 // Compruebo si me da un 200 al hacer la petición
34 if (getStatusConnectionCode(urlPage) == 200) {
35

```

- A continuación, si el servidor es correcto, se accede al programa de *scrapping*, descargando todo el documento HTML de la página en cuestión con el método `getHtmlDocument()` y almacenándolo en el objeto `document` de la clase `Document` de *JSoup*.

```

33 // Compruebo si me da un 200 al hacer la petición
34 if (getStatusConnectionCode(urlPage) == 200) {
35
36     // Obtengo el HTML de la web en un objeto Document
37     Document document = getHtmlDocument(urlPage);
38

```

Una vez se ha obtenido la información, lo que queda es jugar con los nombres y tipos de etiquetas para ir obteniendo la información de dicho documento HTML.

Como el foro consta de varias páginas, toda esta acción se realiza dentro de un bucle `for`, para realizarlo por cada página definida en la URL de entrada.

```

11 public static final String url = "https://www.verema.com/foros/hosteleria/temas?page=%s&tab=foros";
12 public static final int maxPages = 6;
13
14 public static void main( String[] args ) throws Exception{
15
16     PrintWriter out = new PrintWriter(new FileWriter("D:\\outScrapping.html"));
17
18     int numPagina = 0;
19     int numeroEntradas = 0;
20
21     for (int i=1; i<maxPages; i++){
22         Thread.sleep(1000);
23
24         String urlPage = String.format(url, i);
25

```

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

2.3.-Trabajando con las clases y métodos de JSoup.

2.3.1.-Parte 01: Selección de información general.

El foro está formado por 5 páginas iniciales, en las que se describe un pequeño resumen de lo que trata cada foro en particular. Después, una segunda parte interna describe lo que incluye cada post, con cada respuesta de los usuarios sobre ese tema.

The screenshot shows a web browser displaying a forum page. A table lists various topics with columns for the topic name, number of responses, and the latest message. The topics include '¿Cómo afrontar la situación del COVID en hostelería?', 'Rentabilidad de un bar/restaurante', 'Traspaso restaurant', 'Problemas con MBH ???', and 'Distribuidora de vinos italianos'. The developer console on the right shows the HTML structure of the table, highlighting the 'table' and 'tr' elements.

Utilizando el método `select()`, y definiendo las palabras reservadas del tipo de etiqueta y el tipo de clase de cada etiqueta, se va a ir seleccionando de forma jerárquica el contenido hasta obtener la información requerida.

- El objeto `temas` recoge el bloque `table.temas` (etiqueta `table` de clase `temas`). En él se recogen todos los temas por cada página.
- El objeto `trs` selecciona los elementos que incluyan `tr` dentro de ese bloque.
- Y dentro de estos `trs`, `h5` recoge todas las etiquetas que incluyan `h5`, que son los que recogen el enlace y el nombre de cada tema del foro.

```

36 // Obtengo el HTML de la web en un objeto Document
37 Document document = getHtmlDocument(urlPage);
38
39 // Busco todas las entradas que estan dentro de:
40
41 Elements temas = document.select("table.temas");
42 Elements trs = temas.select("tr");
43
44 Elements h5 = temas.select("h5");
45

```

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

Será el tamaño de h5 el que recogerá el número de temas por cada página del servidor, obteniendo el resultado en las salidas:

```

47      System.out.println("\t-->>PAGINA "+numPagina+"<--");
48      out.println("<dd>////////////////////////////////////");
49      out.println("<dd>");
50      out.println("<dd>-->>PAGINA "+numPagina+"<--</dd>");
51      System.out.println("-Número de entradas en página "+numPagina+": "+h5.size()+"\n");
52      out.println("<br>-Número de entradas en página "+numPagina+": "+h5.size()+"<br>");
53      //-----

```

De nuevo un bucle for nos permite obtener toda la información incluida dentro de cada tema, como Título, Autor y Enlace, a través del cual accederemos nuevamente para hacer un *scrapping* más detallado de cada tema.

```

55      // Paseo cada una de las entradas
56      for (Element elem : trs) {
57          Elements entradas = elem.select("h5");
58
59          if(entradas.size() > 0) {
60              Elements elems = entradas.select("a");
61              Element a1 = elems.get(0);
62              String enlace = a1.absUrl("href");
63              String titulo = a1.getElementsByTag("a").text();
64              String autor = elem.getElementsByClass("forum-page-user-nick").text();
65
66              //-----
67              System.out.println("TITULO: "+titulo+"\nAUTOR: "+autor+"\nENLACE: "+enlace+"\n");
68              out.println("<br>TITULO: "+titulo+"<br>AUTOR: "+autor+"<br>ENLACE: "+enlace+"<br><br>");
69              //-----

```

Cada elemento tr tendrá un contenido h5, el cual recogerá enlace (String enlace = a1.absUrl("href");), titulo del tema (String titulo = a1.getElementsByTag("a").text();) y autor (String autor = elem.getElementsByClass("forum-page-user-nick").text();). En la captura de abajo se observa esta información dentro del código HTML del servidor:

Una vez se obtiene el enlace de cada tema (href), este mismo se utiliza para realizar una nueva búsqueda de extracción, pero esta vez ya dentro de cada tema. Como cada tema tiene varias páginas (hay temas que, internamente, tienen hasta 10 páginas de conversación) se tiene que preparar un código mediante bucle for para que recoja toda la información.

Aquí es donde he tenido más problemas, debido a las numerosas solicitudes sobre el servidor cuando ejecutaba el programa. El código lo he dejado comentado en el proyecto (**líneas 138-216**). Es exactamente igual que el que está operativo, pero con el

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

bucle for añadido. Al final tuve que eliminar el bucle for para poder llevar a cabo la actividad. Pero para que quede constancia de que está hecho, lo dejo comentado.

```

138
139
140
141
142
143
144

```

/*
 * Esta parte comentada es para hacer un bucle
 * de todas las páginas de cada hilo.
 * Debido al error 429 de sobresaturación de peticiones lo tuve que obviar.
 * Pero para dar constancia de haberlo hecho, lo dejo comentado en este código.
 */

```

Exception in thread "main" org.jsoup.HttpStatusException: HTTP error fetching URL. Status=429, URL=https://www.verema.com/foros/hosteleria/temas/1479359-todo
at org.jsoup.helper.HttpConnection$Response.execute(HttpConnection.java:762)
at org.jsoup.helper.HttpConnection$Response.execute(HttpConnection.java:707)
at org.jsoup.helper.HttpConnection.execute(HttpConnection.java:297)
at org.jsoup.helper.HttpConnection.get(HttpConnection.java:286)
at DARAct03_WebScraping.WebScraping.App.main(App.java:152)

```

Captura del error citado.

2.3.2.-Parte 02: Selección de información interna de cada hilo.

Una vez dentro de cada tema, se obtiene el contenido HTML nuevamente a través del enlace obtenido en la primera parte. Y así para cada tema.

De este documento HTML, necesitaremos obtener Título, número de respuestas al tema, y todo el contenido de cada hilo del post.

The screenshot shows a web browser displaying a forum post on the website 'verema.com'. The post title is '¿Cómo afrontar la situación del COVID en hostelería?'. The post content includes a user profile for 'Mariagomez' and the text of the post. A red box highlights the title of the post. Another red box highlights the HTML structure in the developer tools, specifically the 'core-Thread_Header' element. An arrow points from the red box in the developer tools to the red box in the browser window, indicating the selection of the title.

-Una primera parte selecciona el contenido para obtener el Título del tema, recogido en la etiqueta `article.core-Thread`.

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

```

78 Document doc = Jsoup.connect(enlace).get();
79
80 Elements mainContent = doc.select("article.core-Thread");
81 Elements header = mainContent.select("header");
82 Element header01 = header.get(0);
83 String title = header01.getElementsByClass("core-Thread_Heading").text();
84

```

-A continuación, necesitamos el número de respuestas, recogidas en la etiqueta core-Thread_Replies.

```

85 Elements respuestas = mainContent.select("div.core-ContainerInHalves_HalfLeft");
86 Elements span = respuestas.select("span");
87 Element res = span.get(0);
88 String respuesta = res.getElementsByTag("span").text();
89

```

-Y, finalmente, un bucle for va extrayendo la información de cada hilo del post, recogiendo todo el contenido en el array de etiquetas div.foros_de_debate.

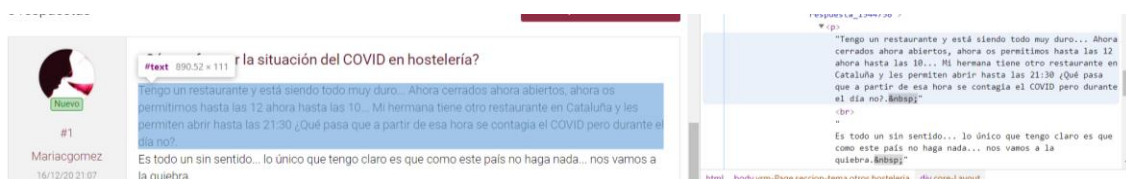
-Dentro del mismo, se obtendrá el número del Post (String numeroPost = numero01.getElementsByTag("a").text();), nombre del autor (String autorPost = autor01.getElementsByTag("span").text();), fecha (String fechaPost = elemento.getElementsByTag("time").text();) y contenido íntegro (String contenidoPost = parrafo.getElementsByTag("p").text();).

```

95 Elements forosDebate = mainContent.select("div.foros_de_debate");
96 Elements foros = forosDebate.select("article.core-Post");
97 for(Element elemento : foros) {
98     Elements numero = elemento.select("div.core-Spacer-top-medium");
99     Elements numeroA = numero.select("a");
100     Element numero01 = numeroA.get(0);
101     String numeroPost = numero01.getElementsByTag("a").text();
102
103     Elements auto = elemento.select("a.core-UserPane_Username");
104     Elements autorA = auto.select("span");
105     Element autor01 = autorA.get(0);
106     String autorPost = autor01.getElementsByTag("span").text();
107
108     String fechaPost = elemento.getElementsByTag("time").text();
109
110     //-----
111     System.out.println("\t-Post "+numeroPost+"\n\t-Autor: "+autorPost+"\n\t-Fecha: "+fechaPost);
112     out.println("<dd>-Post "+numeroPost+"<br>-Autor: "+autorPost+"<br>-Fecha: "+fechaPost+"<br>");
113     //-----
114
115     Elements contenido = elemento.select("div.core-PostMessage");
116     Elements contenidoA = contenido.select("main");
117     Elements parrafos = contenidoA.select("p");
118

```

-Para los contenidos, se observa en el código HTML que vienen definidos por párrafos (<p></p>), con lo cual se tiene que hacer una selección dentro de un contenido con etiquetas div.core-PostMessage, e ir obteniendo cada párrafo mediante un bucle for.



Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

```

115         Elements contenido = elemento.select("div.core-PostMessage");
116         Elements contenidoA = contenido.select("main");
117         Elements parrafos = contenidoA.select("p");

124         for(Element parrafo : parrafos) {
125             String contenidoPost = parrafo.getElementsByTag("p").text();
126             //-----
127             System.out.println("\t\t"+contenidoPost);
128             out.println("<dd><p>"+contenidoPost+"</p></dd>");
129             //-----

```

Durante el desarrollo de la implementación, utilicé el método de salida por consola `System.out.println()` para ir comprobando la información obtenida.

```

16         PrintWriter out = new PrintWriter(new FileWriter("D:\\outScraping.html"));

```

Pero para trasladar dicha información a un documento HTML, utilicé la clase `PrintWriter` para crear el objeto `out`, con el fin de que el programa vaya escribiendo toda la información en el documento HTML `outScraping.html` a medida que se va desarrollando. De ahí que, el método `out.println()` de la clase `PrintWriter`, comparta espacio con `System.out.println()`, para que obtengan la misma información en la misma posición, tanto de la consola como del documento.

```

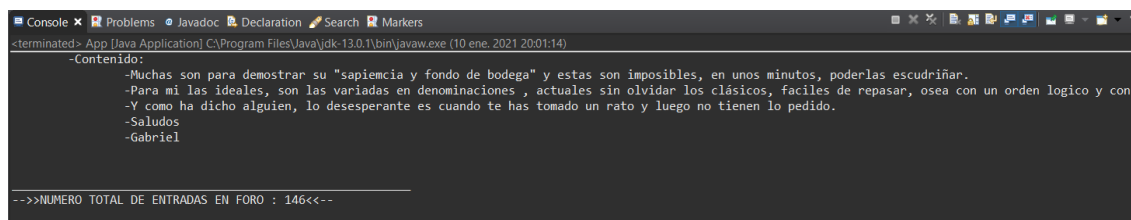
46         //-----
47         System.out.println("\t-->>PAGINA "+numPagina+"<--");
48         out.println("<dd>////////////////////////////////////");
49         out.println("<dd>");
50         out.println("<dd>-->>PAGINA "+numPagina+"<--</dd>");
51         System.out.println("-Número de entradas en página "+numPagina+": "+h5.size()+"\n");
52         out.println("<br>-Número de entradas en página "+numPagina+": "+h5.size()+"<br>");
53         //-----

229         //-----
230         System.out.println("_____"");
231         out.println("<dd>_____"</dd><br>");
232         System.out.println("-->>NUMERO TOTAL DE ENTRADAS EN FORO : "+numeroEntradas+"<--\n");
233         out.println("<dd>-->>NUMERO TOTAL DE ENTRADAS EN FORO : "+numeroEntradas+"<--</dd><br><br>");
234         //-----
235
236         out.close();

```

2.4.-Presentación de los posts (HTML y Javascript).

Una vez se ejecuta el programa y se obtiene toda la información “scrappeada”, realizo el proyecto web desde un documento HTML de forma similar a las actividades anterior, subiendo todo su contenido a un repositorio *GitHub*.



```

-Contenido:
-Muchas son para demostrar su "sapiencia y fondo de bodega" y estas son imposibles, en unos minutos, poderlas escudriñar.
-Para mí las ideales, son las variadas en denominaciones , actuales sin olvidar los clásicos, fáciles de repasar, osea con un orden logico y con a
-Y como ha dicho alguien, lo desesperante es cuando te has tomado un rato y luego no tienen lo pedido.
-Saludos
-Gabriel

-->>NUMERO TOTAL DE ENTRADAS EN FORO : 146<--

```

Captura de una parte de la salida por consola de los resultados del *scraping*.

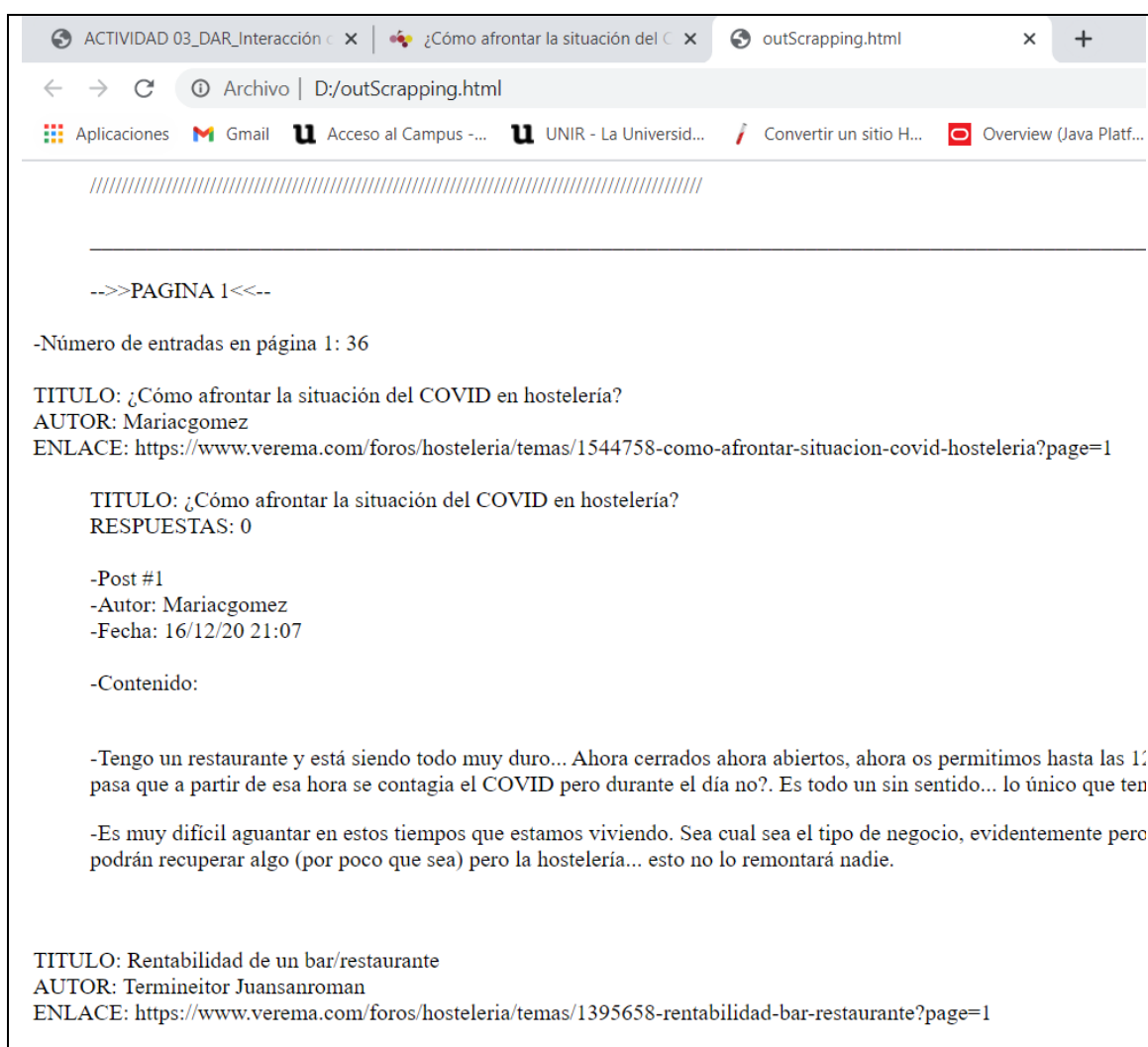
Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

Para que el documento HTML recoja la información de forma indexada, utilicé las directivas propias de HTML escritas dentro de los parámetros (Strings) de información en el método `out.println()`. Una prueba de ello es la siguiente captura:

```
127         System.out.println("\t\t-"+contenidoPost);
128         out.println("<dd><p>-"+contenidoPost+"</p></dd>");
129         //
```

Uso de las directivas `<dd>
` y `<p>` en las salidas de contenido HTML.

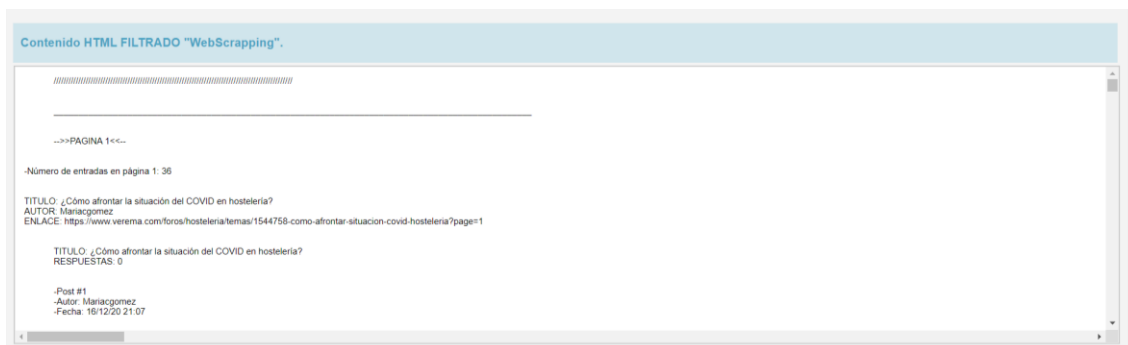
El resultado, es el documento HTML `outScraping.html`, con todo el contenido indexado:



2.4.1.-Presentación del Proyecto.

La idea original era hacer partícipe al usuario para que obtuviera el documento *scrappeado* al hacer click sobre un botón. De ahí que, utilizando la base HTML de la estética de las antiguas actividades de la asignatura (actividad 02), usara una función

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	



Para finalizar la presentación, realicé una Descripción de la actividad (resumen de esta memoria) en formato HTML, como un anexo del proyecto de presentación (anexo al *index.html*), de tal forma que se pueda ver explicado en el proyecto web.

Esta memoria también se puede descargar desde la cabecera del documento HTML de la propia página web, además del Proyecto Java en formato .zip.



2.5.-Conclusión.

Aunque hemos pasado por varias fases de ejecución de la actividad, por los diferentes problemas que hemos tenido con el foro con *login*, superando estas adversidades he de decir que la actividad me ha gustado y la he disfrutado. Además de estos problemas, he tenido otros, descritos en esta memoria, relativos a las conexiones y los estados Http del propio servidor (*Status 429*).

No ha sido fácil cogerle el truco a la librería *JSoup*, ya que cada servidor web es diferente y hay que tener muy claro el contenido a seleccionar. Cada código HTML tiene un nombre en las etiquetas y las clases, y la distribución jerárquica es diferente.

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

En este foro en particular, algo que ha complicado el *scrapping* ha sido la distribución en varias páginas de los diferentes temas. Lo que me ha llevado a ir buscando alternativas de implementación en *Java* sobre la marcha.

Por último, al igual que especifiqué el código comentado en el proyecto y las razones por las cuales lo he dejado, entre las líneas 243 y 309 se encuentra desarrollado el método `posts()`. En un principio lo desarrollé con la idea de no tener que escribir este código en el método principal, y poder llamarlo cada vez que se necesitara. Pero, al tener que extraer la información al documento HTML, `PrintWriter` no recogía la información de un método definido externamente al método principal donde se ha instanciado dicha clase `PrintWriter`. De ahí que lo obvié. Y dicho código está resuelto dentro del método principal. Pero para que quede constancia de haberlo llevado a cabo, lo he mantenido tal cual en el código.

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

3.-Referencias. Bibliografía. Webgrafía.

3.1.-Maven y Scrapping en Eclipse.

Ramón Invarato (13 de enero de 2015). **jarroba.com**. *Maven en Eclipse*. Recuperado el 6 de enero de 2021 de <https://jarroba.com/maven-en-eclipse/>

Ricardo Moya (12 de mayo de 2015). **jarroba.com**. *Scraping en Java (JSoup), con ejemplos*. Recuperado el 6 de enero de 2021 de <https://jarroba.com/scraping-java-jsoup-ejemplos/>

3.2.-JSoup.

Jonathan Hedley (2009-2021). **jsoup.org**. *Descarga e Instalación de JSoup*. Recuperado el 6 de enero de 2021 de <https://jsoup.org/download>

Sundhar (User StackOverrun) (10 de noviembre de 2010). **stackoverflow.com**. *How to extract absolute URL from relative HTML links using Jsoup?*. Recuperado el 7 de enero de 2021 de <https://stackoverflow.com/es/q/972135>

TopBlog (StackOverrun user) (25 de febrero de 2012). **stackoverflow.com**. *Extract Span tag data using Jsoup*. Recuperado el 7 de enero de 2021 de <https://stackoverflow.com/es/q/2472227>

Matt9Atkins (StackOverrun user) (27 de mayo de 2013). **stackoverflow.com**. *Java - Obtain text within script tag using Jsoup*. Recuperado el 7 de enero de 2021 de <https://stackoverflow.com/es/q/4564493>

Living Sun (2018). **living-sun.com**. *¿Cómo obtener el texto de la clase div? - java, html-parsing, jsoup*. Recuperado el 7 de enero de 2021 de <https://living-sun.com/es/java/471038-how-to-get-the-text-from-div-class-java-html-parsing-jsoup.html>

Asignatura	Datos del alumno	Fecha
Desarrollo de Aplicaciones en Red	Apellidos: Gil Valencia	21 de enero de 2021
	Nombre: José Antonio	

3.3.-Java.

Cs Carleton (sin fecha). **cs.carleton.edu**. *Comandos Java de ejecución para salida de la información por Consola*. Recuperado el 7 de enero de 2021 de <https://www.cs.carleton.edu/faculty/dmusican/cs117s03/iocheat.html>

Guillermo (Stackoverflow user) (22 de junio de 2017). **cs.stackoverflow.com**. *Enviar un string con un (salto de línea) por request en java*. Recuperado el 7 de enero de 2021 de <https://es.stackoverflow.com/questions/80802/enviar-un-string-con-un-salto-de-l%C3%ADnea-por-request-en-java>

DarkFenix (Stackoverflow user) (16 de noviembre de 2016). **cs.stackoverflow.com**. *¿ Como eliminar los n últimos caracteres de una cadena en Java?*. Recuperado el 8 de enero de 2021 de <https://es.stackoverflow.com/questions/34254/como-eliminar-los-n-%C3%BAltimos-caracteres-de-una-cadena-en-java-java>

3.4.-Salida de Java a archivo HTML.

Colin (user Stackoverflow) (9 de junio de 2016). **cs.stackoverflow.com**. *Console output to HTML File*. Recuperado el 8 de enero de 2021 de <https://stackoverflow.com/questions/37729992/console-output-to-html-file>

3.5.-AJAX.

Jose Aguilar (24 de septiembre de 2018). **jose-aguilar.com**. *Abrir contenido sin recargar la página*. Recuperado el 9 de enero de 2021 de https://www.jose-aguilar.com/blog/abrir-contenido-sin-recargar-la-pagina/#codesyntax_3

Codingraph (Youtube user) (24 de septiembre de 2018). **youtube.com**. *Cargar contenido HTML usando AJAX*. Recuperado el 9 de enero de 2021 de <https://www.youtube.com/watch?app=desktop&v=BP5LesQGns8>