

Prácticas de Laboratorio

Nota: El lenguaje de programación recomendado para el desarrollo de las prácticas es Java. Para la entrega de cada bloque, se hará en un zip con el código y además una memoria donde debe aparecer para cada ejercicio los siguientes puntos:

- Descripción de la tarea realizada
- Capturas de pantalla de la ejecución
- Conclusiones

BLOQUE Nº 2. Programación con sockets**I. Desarrollo de servicios web básicos en Java.**

1. Codificar un pequeño cliente HTTP que abra una conexión con el servidor <http://www.lcc.uma.es>, inicie la descarga del fichero cuyo nombre se le pasa como parámetro, lo almacene localmente y muestre el resultado de llamar a los siguientes métodos:

- `getContentLength()`
- `getContentType()`
- `getDate()`
- `getLastModified()`

2. Desarrollo de un servidor web simple en Java de acuerdo a las siguientes especificaciones:

- Debe ser un servidor concurrente TCP y debe esperar conexiones en el puerto 8000
- Solamente soporta la gestión de peticiones GET
 - Ej: GET objeto HTTP/1.1
- No soporta cookies ni sesiones
- Solamente soporta las respuestas:
 - HTTP/1.0 200 OK
 - `<html><body><h1>404 Not Found</h1></body></html>`
 - HTTP/1.0 404 Not Found
 - HTTP/1.0 404 Permission Denied
- Los tipos MIME que soporta son:
 - `text/plain` (correspondiente a ficheros `.txt`)
 - `text/html` (correspondiente a ficheros `.htm` y `.html`)
- Debe enviar las siguientes cabeceras:
 - `Content-Type`
 - Ej: `Content-Type: text/html`
 - `Content-Length`
 - Ej: `Content-Length: 3453`

II. Desarrollo de servicios web con servlets y páginas JSP.

1. Probar los *servlets* vistos en clase en las diferentes transparencias.

2. Copiar y editar el ejemplo con *Login* visto en clase, cuyo código se incluye en el campus virtual y ejecutarlo, tanto la implementación con *servlets* como la implementación con páginas JSP. Este ejercicio pretende que te sirva de base para realizar el ejercicio obligatorio.

- Diseña tu propia solución aplicando el patrón de diseño MVC y utilizando *servlets*, *beans* y páginas JSP para implementarlo. (si decides hacerlo así para el ejercicio obligatorio).
- Agrégale la autenticación con contraseña
- Busca la cookie que genera esta aplicación con la sesión que has creado

3. Se pretende desarrollar un **sistema de acceso a una biblioteca digital**. Este sistema permitirá descargar documentos como artículos o libros desde la página web a los usuarios que estén registrados y se hayan identificado correctamente. Se implementará usando *servlets*, proporcionando la siguiente funcionalidad:

- La página inicial mostrará un formulario donde se podrá introducir el nombre de usuario y la contraseña, así como un enlace a la función de registro de nuevo usuario. Se validará el formulario como en el ejercicio anterior.
- La pantalla de registro mostrará un formulario para introducir el nombre de usuario y la contraseña que desee el usuario (y algún campo adicional como nombre o clave de recuperación de cuenta). Cuando el usuario rellene estos datos y envíe el formulario, se almacenará en la lista de usuarios registrados. Sin embargo, ha de comprobarse que no se introduce un nombre de usuario o contraseña nulos y que el nombre de usuario introducido no existe con anterioridad en el sistema. Si se ha registrado correctamente, se le redireccionará a la pantalla de inicio para que pueda iniciar sesión.
- Los usuarios registrados se almacenarán en un fichero de texto (llamarlo “usuarios”) en el que cada línea contendrá el nombre de usuario y su contraseña tal y como se indica en el ejemplo:

```
usuario1 password1
usuario2 password2
usuario3 password3
```

Para introducir la contraseña se deberá usar un campo en el formulario HTML como el siguiente:

```
<input type="password" name="password" />
```

- Cuando el usuario haya iniciado la sesión correctamente, se le mostrará la página inicial de la *Biblioteca digital*. En la parte superior de la página se pondrá un enlace que le permita al usuario cerrar la sesión e información sobre el usuario actual (ej: Usuario: Alicia). A continuación, se mostrará una lista de los documentos existentes en el repositorio de documentos
- El repositorio de la *Biblioteca digital* se puede implementar mediante un fichero de texto con la lista de documentos disponibles (llamar a este fichero documentos) y tendría el siguiente formato:
<titulo><CRLF><autor><CRLF><nombre_fichero><CRLF><resumen><CRLF><CRLF>:

```
Titulo1
Juan Pinto
titulo1.txt
```

resumen1blalblalblalbal
Titulo2
Laura Pinilla
titulo2.pdf
resumen2blalblalblalbal

- En la página de la *Biblioteca digital*, se mostrará la lista de documentos disponibles. Para cada documento se mostrará el título, el autor, el resumen, y un enlace para poder descargar el fichero asociado, si el usuario ha iniciado la sesión correctamente. Si el usuario no se ha autenticado (no se ha creado la sesión), se mostraría el resumen asociado al documento (en la misma o en otra página), pero no el enlace de descarga del fichero. En vez de dicho enlace, habría que incluir algún enlace para redireccionarle a la pantalla de *login* para que el usuario pueda autenticarse y si tiene éxito redirigirlo a la misma página donde había seleccionado el documento que quería descargarse.
- También incluir en la página de la *Biblioteca digital* la búsqueda de documentos por el campo Título o Autor (elegir uno). Una vez establecido el patrón de búsqueda, se devolverá un listado de los documentos que cumplan dicho patrón, todos en una misma página.
- Probar la práctica incluyendo al menos diez documentos.

Nota: No usar código javascript, ni eventos/objetos del DOM (ej: onClick), ni hojas de estilo CSS ni etiquetas de formato, que no sean las básicas de HTML (ej: , , <h1>, ...), ya que estas tecnologías se aplicarán en la siguiente práctica.

4. Ejercicio 3 ampliado. Se podrá mejorar y extender el funcionamiento básico del ejercicio 3 considerando una o varias de las sugerencias que se hacen a continuación.

- Aplicar el MVC e implementar las vistas usando JSP en lugar de *servlets*.
- Agregar una búsqueda avanzada que permita buscar documentos por más de un campo (ej: Título="El principito" y Autor="Saint-Exupéry"). Una vez establecido el criterio de búsqueda, se devolverá un listado de los documentos que cumplan dicho criterio que incluirá el título, autor, enlace y resumen para cada uno de ellos, dependiendo de si el usuario se ha autenticado o no.
- Almacenar el número de veces que se descarga cada documento y mostrar esta información en los listados de documentos.
- Mostrar solamente 5 documentos por página e incluir enlaces para avanzar a las siguientes páginas 2, 3, ...
- Cualquier otra mejora que se consideres apropiada

Obligatorios: II.3

Optativo: I.2 y II.4 (en este caso, si se entrega en fecha, entregar solamente el ampliado e indicar en la memoria con qué características se ha ampliado)

Fecha entrega: 30/11/22 (para el obligatorio)