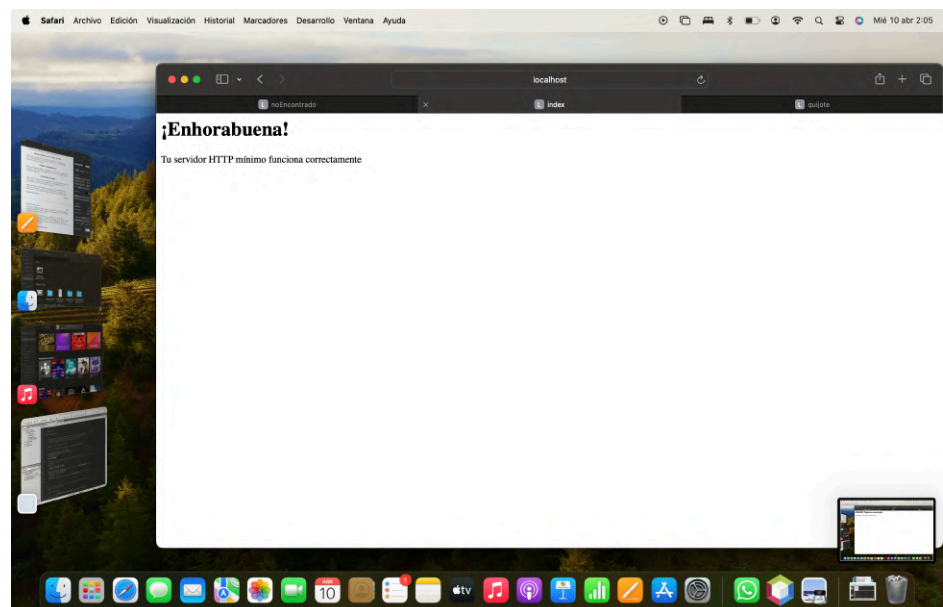
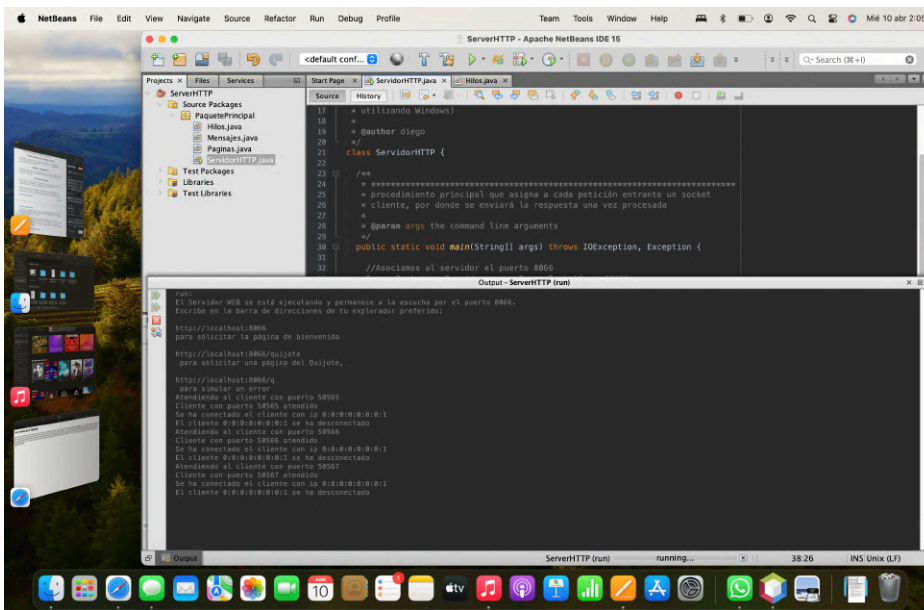


# Tarea 5 para Programación de Servicios y Procesos



## Diego Manuel Carrasco Castañares

**Ejercicio.**

Crearemos un servidor HTTP similar al del ejemplo del servidor HTTP (Proyecto java ServerHTTP, apartado 5.1 de los contenidos).

Pero un servidor HTTP realista tendrá que atender varias peticiones simultáneamente. Para ello, tenemos que ser capaces de modificar su código para que pueda utilizar varios hilos de ejecución. Esto se hace de la siguiente manera:

- El hilo principal (o sea, el que inicia la aplicación) creará el socket servidor que permanecerá a la espera de que llegue alguna petición.
- Cuando se reciba una, la aceptará y le asignará un socket cliente para enviarle la respuesta. Pero en lugar de atenderla él mismo, el hilo principal creará un nuevo hilo para que la despache por el socket cliente que le asignó. De esta forma, podrá seguir a la espera de nuevas peticiones.

Por lo tanto modifica el ejemplo del servidor HTTP (Proyecto java ServerHTTP, apartado 5.1 de los contenidos) para que implemente multihilo, y pueda gestionar la concurrencia de manera eficiente.

**Criterios de puntuación. Total 10 puntos..Se tendrá en cuenta:**

- El funcionamiento correcto del programa que se piden.
- El uso adecuado de las API.
- Tratamiento adecuado de posibles excepciones.

**Recursos necesarios para realizar la Tarea.**

Contenidos y ejemplos de la unidad, en especial los apartados 5.1.- Programación de un servidor HTTP y el 5.2.- Implementar comunicaciones simultáneas.

**Consejos y recomendaciones.**

Realiza previamente todos los ejemplos de la unidad y pruébalos en tu equipo local. Para la prueba de algunos de ellos, debes tener conexión a Internet.

**Indicaciones de entrega.**

Debes crear un único fichero comprimido en el que debes incluir el o los proyectos Netbeans y documentos que correspondan a la realización de los ejercicios de la tarea.

Además adjuntarás un documento realizado con un procesador de texto con unas breves explicaciones de cómo has realizado los ejercicios, con las capturas de pantallas que estimes oportunas.

El fichero comprimido debe seguir la nomenclatura PSPXX\_TareaZZ\_apellido1\_apellido2\_nombre, donde XX se corresponde con la unidad y ZZ con el numero de la tarea.

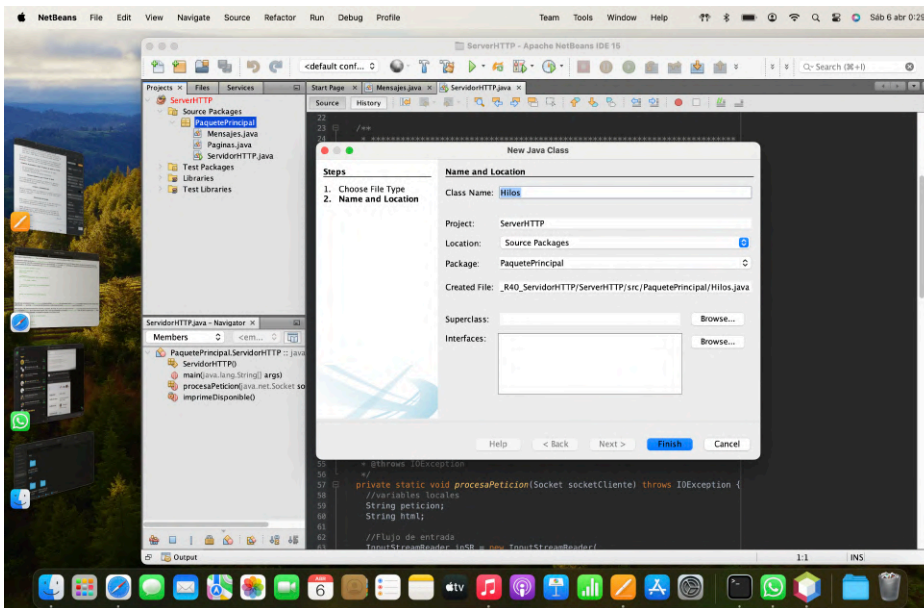
Realiza el envío de dicho archivo comprimido a través de la plataforma. Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna Begoña Sánchez Mañas para la primera unidad del MP de PSP, debería nombrar esta tarea como..

sanchez\_manas\_begona\_PSP05\_Tarea

**Indice:**

Clase Hilo.....	01
Clase ServidorHTTP.....	01
Capturas del funcionamiento.....	01

Empezaremos descargando el proyecto de ejemplo de la unidad. Tras ello vamos a crear una nueva clase llamada Hilos.



A dicha clase le diremos que extiende de Thread.

En ella vamos a declarar dos variables, un Socket de tipo privado y un ServidorHTTP (clase que contiene el método main en el proyecto) también de tipo privado.

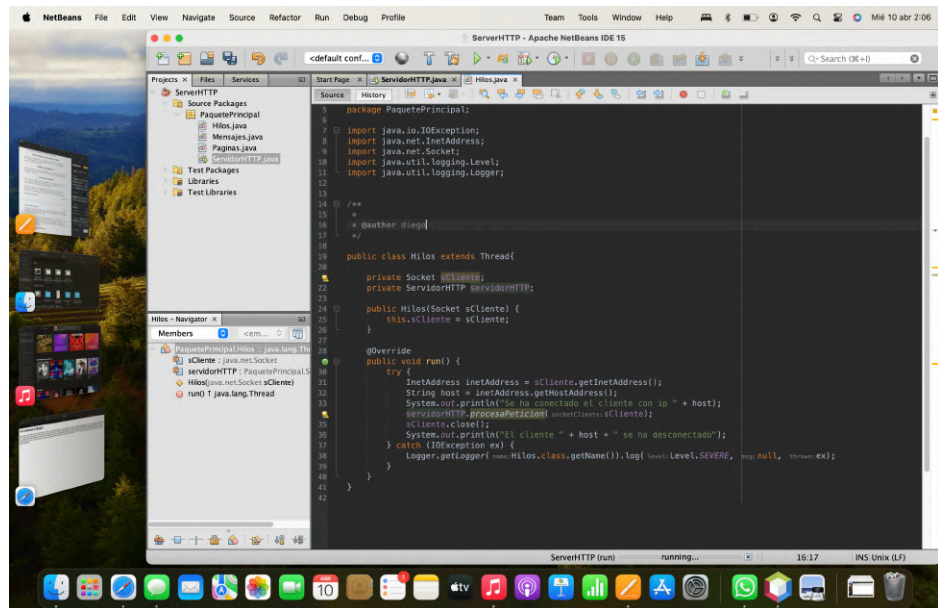
En el constructor por defecto le pasaremos por parámetro el Socket del cliente.

En el método run declararemos un InetAddress, al cual le vamos a pasar, también, el Socket del cliente.

Tras ello declaramos un string que se instanciará con el inetAddress anterior.

Posteriormente mostraremos por pantalla que se he conectado un cliente, cerraremos la conexión y mostraremos por pantalla que se ha desconectado dicho cliente.

Todo ello irá envuelto en try - catch.



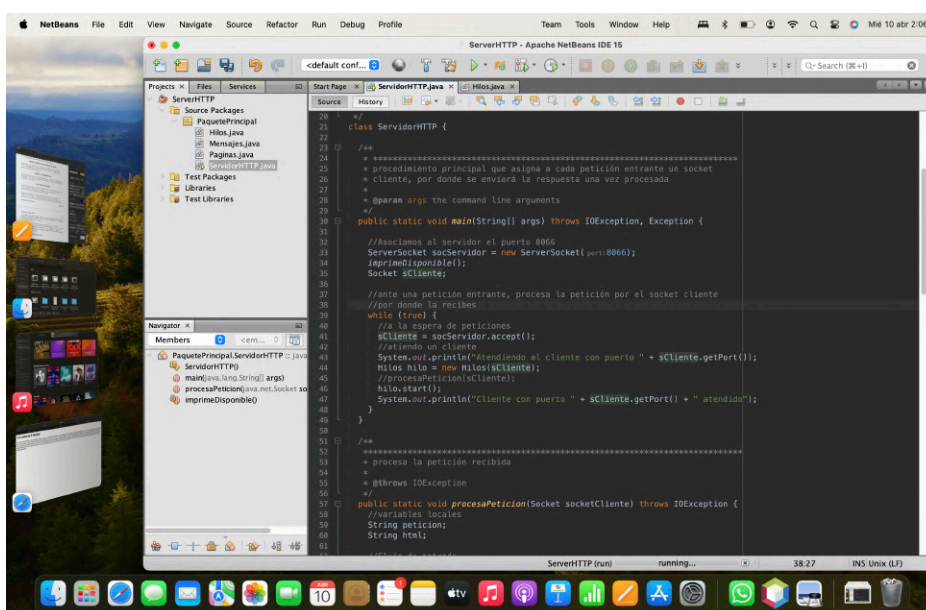
La clase ServidorHTTP solo será modificada en su método main.

Dentro de dicho main está declarado un while, con condición true.

Dentro de este while aceptaremos el socket del servidor, mostraremos por pantalla que se ha conectado el cliente.

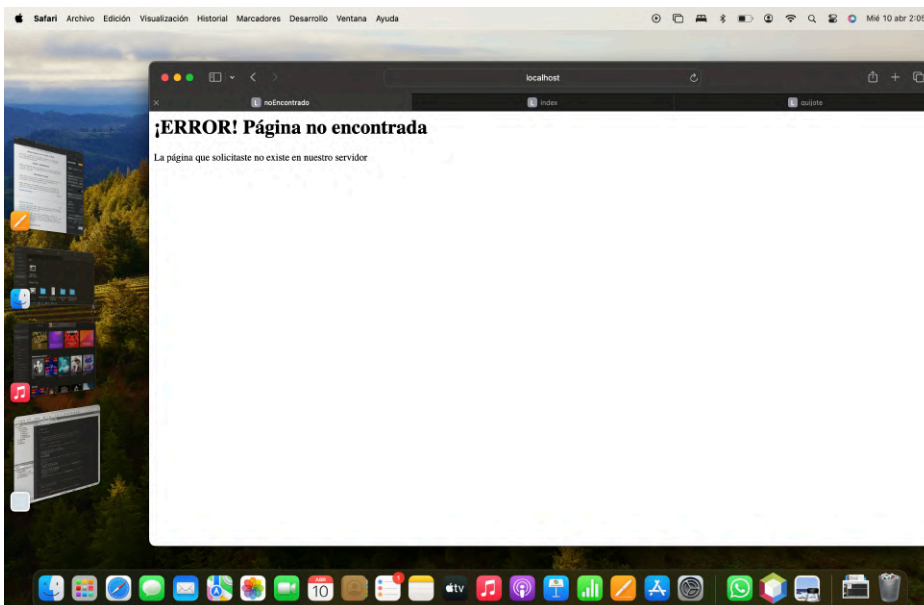
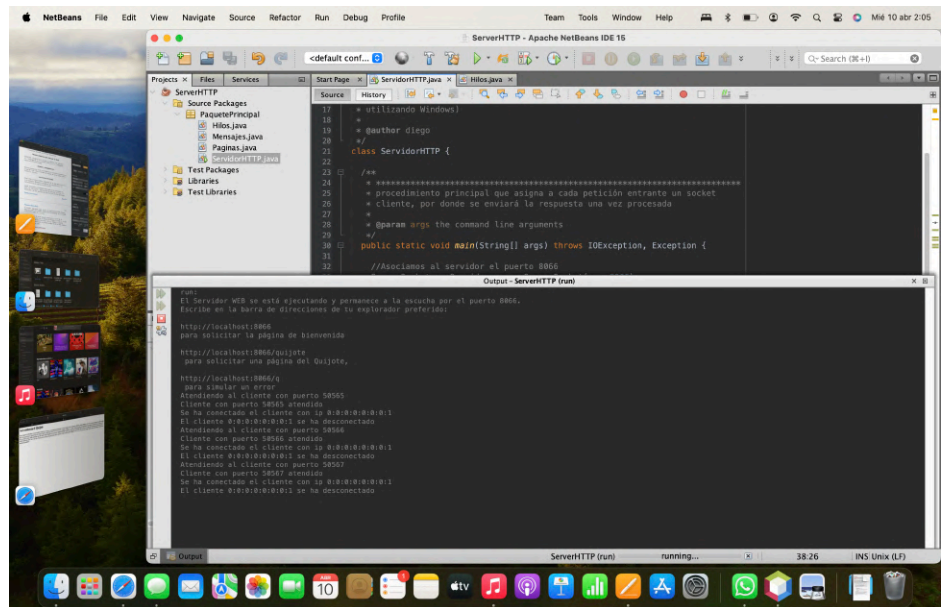
Tras ello declararemos e instanciaremos un objeto de la clase Hilos pasándole el socket del cliente e iniciaremos el hilo mediante el método start y mostraremos por pantalla que el cliente ha sido atendido.

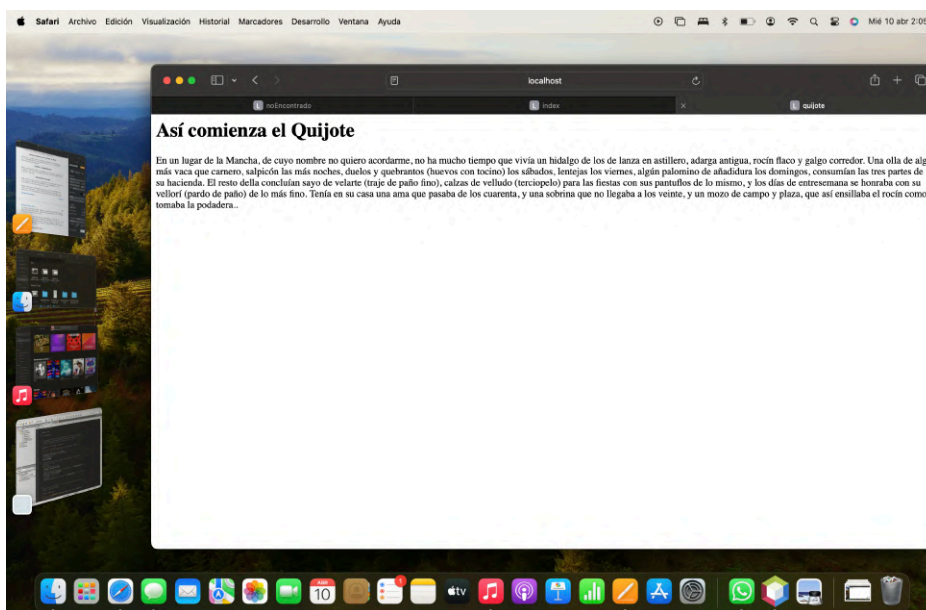
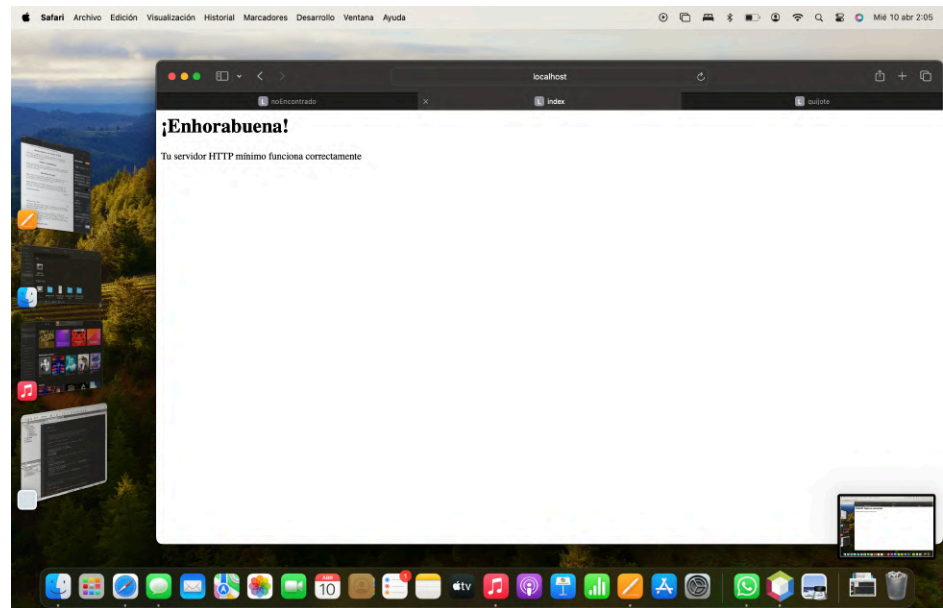
Todo lo demás de esta clase y las demás que contiene el proyecto de ejemplo no serán modificadas.





En las siguientes imágenes podemos ver el output de NetBeans y las páginas abiertas en el navegador.





Y con esto damos por finalizada la tarea.