

Actividad

Actividad AE5_T5_ServiciosRed

Ubicación

Tema 5 - Generación de servicios en red

Objetivos

- Aprender a crear un servidor HTTP que pueda gestionar peticiones GET y POST.
- Integrar en el servidor un cliente de correo electrónico que permita el envío automático.

Temporalización

La duración prevista para esta actividad es de seis sesiones lectivas.

Instrucciones

Después que sus consejeros le dijeran aquello de “winter is coming”, Lord Eddard Stark ha decidido hacer una pequeña inversión y va a instalar algunas estufas de pellets en Invernalía. El modelo que le han recomendado lleva incorporado un pequeño servidor HTTP que permite el control remoto, pero el software de control no convence a Lord Stark y decide encargar un desarrollo en Java a algún técnico superior de DAM del reino.

Esto es lo que se indica en el pergamino de especificaciones:

- El servidor (Servidor.java) estará alojado en la IP 127.0.0.1 en el puerto 7777.
- El servidor aceptará peticiones a través del navegador en la ruta de contexto “/estufa”.
- El servidor será multihilo y gestionará las peticiones desde la clase GestorHTTP.java.
- La clase GestorHTTP tendrá dos atributos: temperaturaActual y temperaturaTermostato. Inicialmente ambas temperaturas tendrán el mismo valor, por ejemplo 15º.
- El gestor HTTP de contexto puede proporcionar como GET el dato “temperaturaActual” (ejemplo: <http://localhost:7777/estufa?temperaturaActual>). Deberá “servir” al cliente una página HTML (formato libre) donde se vean ambas temperaturas (actual y termostato).
- El gesto HTTP aceptará como POST la instrucción “setTemperatura=X”, donde X es la temperatura que debe alcanzar la estufa (temperaturaTermostato). Deberá mostrar un mensaje informativo por pantalla en la aplicación Java y otro en el cliente Postman.
- Aparte de los métodos para gestionar los GET y POST, la clase GestorHTTP tendrá un método llamado “regularTemperatura()”. Este método se encargará de variar (aumentar o disminuir) la temperaturaActual de la estufa en 1 unidad hasta alcanzar la temperaturaTermostato que se haya pasado como parámetro POST. Para que el proceso de variación dure algo más de tiempo y se pueda ir consultando desde el navegador, habrá una pausa de 5 segundos entre cada aumento/disminución de la temperatura.

Ampliación:

El único técnico capaz de reparar estufas de pellets cometió algún pequeño delito y ahora está en la Guardia de la Noche, destinado en el Muro.

Para no quedarse sin servicio, Lord Stark te ofrecerá una succulenta recompensa (2 puntos) si implementas un sistema de alerta que permita enviar automáticamente un email al técnico.

Además, debes anexar dos documentos (una imagen y un PDF) y poner en copia a Lord Stark, para que vea lo bien que has trabajado.

Consideraciones a la ampliación:

- El gestor HTTP aceptará como POST la siguiente instrucción, que deberá activar el envío:

`notificarAveria:email_remitente=EMAIL;pass_remitente=PASSWORD`

donde **EMAIL** debe ser un correo en uso y **PASSWORD** su contraseña.
- El servidor tendrá implementado directamente en su código la información del servidor SMTP de Gmail (smtp.gmail.com), el puerto TLS (587), el asunto (AVERIA), el mensaje (texto libre), la ruta de la imagen a anexar (imagen libre), la ruta del PDF a anexar (PDF libre) y los e-mails del técnico y de Lord Stark.
- El correo electrónico del técnico: mantenimientoinvernalnia@gmail.com
- El correo electrónico (no oficial) de Lord Stark: megustaelfresquito@gmail.com
- Para hacer pruebas de envío desde un e-mail remitente se recomienda crear y utilizar una cuenta de correo de Gmail que no sea personal. Debes configurarla siguiendo las indicaciones dadas en el tema para que pueda aceptar envíos de tu aplicación Java. Para comprobar que llegan los mensajes a utiliza alguna cuenta tuya como destinatario (puede ser Gmail u otras), ya que a las cuentas arriba indicadas sólo tendrá acceso el profesor.

NOTA: en un escenario real no enviaríamos datos como el e-mail o la contraseña en un POST de una conexión HTTP, sino que deberíamos utilizar una conexión HTTPS.

Observaciones

- Se entregará en Florida Oberta un fichero ZIP que incluya el código completo de la actividad, los anexos del correo (si se ha hecho la ampliación) y el enlace al repositorio Github (en un fichero txt).
- La actividad también deberá subirse a una rama de Git "AE<N>" que partirá de "master", donde "N" es el número de actividad.

Evaluación

La actividad es obligatoria. Para la evaluación se tendrá en cuenta el funcionamiento de los programas, la codificación adecuada y la documentación. Se puede solicitar al alumno que explique parte de su código así como que realice pequeñas modificaciones.

Recursos

Material de módulo (Florida Oberta).