

DOCUMENTO DE REQUISITOS Y DISEÑO: PRÁCTICA 1

AUTORES

CARLOS GARCÍA SANTA

EDUARDO JUNOY ORTEGA

REDES DE COMUNICACIONES II

GRUPO 2321: PAREJA 04

INGENIERÍA INFORMÁTICA
ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD AUTÓNOMA DE MADRID



12/03/2024

Índice

Índice.....	2
Introducción.....	3
Alcance del producto.....	3
Objetivos.....	3
Uso.....	3
Definiciones y acrónimos.....	3
Requisitos funcionales y del sistema.....	4
Funciones If/Then.....	4
Lógica de Manejo de Datos.....	4
Requisitos de Desempeño.....	4
Requisitos no funcionales.....	4
Fiabilidad y Disponibilidad.....	4
Facilidad de Uso.....	4
Requisitos de la interfaz externa.....	5
Interfaz de administración del servidor.....	5
Interfaz de hardware.....	5
Interfaz de software.....	5
Interfaz de comunicación.....	5
Diagrama Entidad-Relación.....	5
Casos de uso.....	6
Procesamiento de una Solicitud GET.....	6
Ejecución de un script mediante una Solicitud POST.....	6
Manejo de Solicitudes Concurrentes.....	6
Configuración del Servidor Mediante Archivo de Configuración.....	7
Respuesta a una Solicitud con Método HTTP no Soportado.....	7
Conclusiones personales y técnicas.....	7

Introducción

Alcance del producto

La función de esta práctica es diseñar e implementar un servidor web básico en lenguaje C. Este servidor está diseñado para comprender el protocolo HTTP, junto a conceptos de red y programación de servidores web.

Objetivos

La práctica ha tenido un resultado satisfactorio, ya que cumple los tres objetivos especificados:

- Implementar soporte para los verbos HTTP GET, POST, y OPTIONS.
- Permitir la configuración del servidor a través de un archivo.
- Soportar la ejecución de scripts, añadiendo dinamismo al contenido servido.

Uso

El uso de esta práctica incluye:

- La experimentación y el aprendizaje con servidores web y el protocolo HTTP.
- Desarrollo de habilidades de programación en C en redes de comunicaciones.
- Pruebas de red con distintos tipos de servidores y configuraciones
- Aprendizaje en cuanto a rendimiento y gestión de conexiones concurrentes.

Definiciones y acrónimos

- HTTP (Hypertext Transfer Protocol): Protocolo de comunicación que permite la transferencia de información a través de archivos en la World Wide Web.
- CGI (Common Gateway Interface): Tecnología de la World Wide Web que permite a un cliente solicitar datos de un programa ejecutado en un servidor web.
- IDE (Integrated Development Environment): Un entorno de desarrollo integrado es una aplicación de software que proporciona instalaciones integrales para el desarrollo de software.

Requisitos funcionales y del sistema

Funciones If/Then

- Si el servidor recibe una solicitud válida la procesa según el método (GET, POST, OPTIONS) y recurso solicitado. Devuelve el código de estado 200 (OK)
- Si el servidor recibe una solicitud con una sintaxis errónea, el servidor responde con un código de estado 400 (Bad Request).
- Si el servidor recibe una solicitud para la cual no ha encontrado el recurso, devuelve el código 404 (Not Found).

Lógica de Manejo de Datos

El servidor es capaz de analizar las peticiones HTTP, extrayendo el método HTTP, la URI y cualquier dato enviado por el cliente, así como la gestión de contenido dinámico. Si la solicitud requiere la ejecución de scripts, el servidor debe ejecutar el script y devolver la salida del script como respuesta.

Requisitos de Desempeño

- El servidor devuelve las solicitudes dentro de un umbral de tiempo aceptable.
- Es escalable, es decir, mantiene un rendimiento estable y eficiente a medida que aumenta el número de solicitudes.

Requisitos no funcionales

Fiabilidad y Disponibilidad

El servidor debe ser capaz de operar continuamente minimizando el tiempo de inactividad mediante mecanismos de recuperación ante fallos. De la misma forma implementa herramientas que alertan de errores.

Facilidad de Uso

La administración del servidor hace que la configuración, administración y diagnóstico puedan ser fácilmente configurables, haciendo el sistema accesible para personas sin amplios conocimientos en redes de comunicaciones.

Requisitos de la interfaz externa

Interfaz de administración del servidor

La administración del servidor se realiza en línea de comandos, aunque se pueden realizar pruebas con la visualización desde el navegador web, que en nuestro caso hemos realizado en alguna ocasión.

Interfaz de hardware

El servidor es capaz de operar en hardware de red estándar siguiendo los protocolos de red pertinentes. Además, el servidor no tiene requerimientos mínimos del sistema, pudiendo utilizar prácticamente cualquier CPU, memoria RAM y espacio de almacenamiento disponible en el mercado actual.

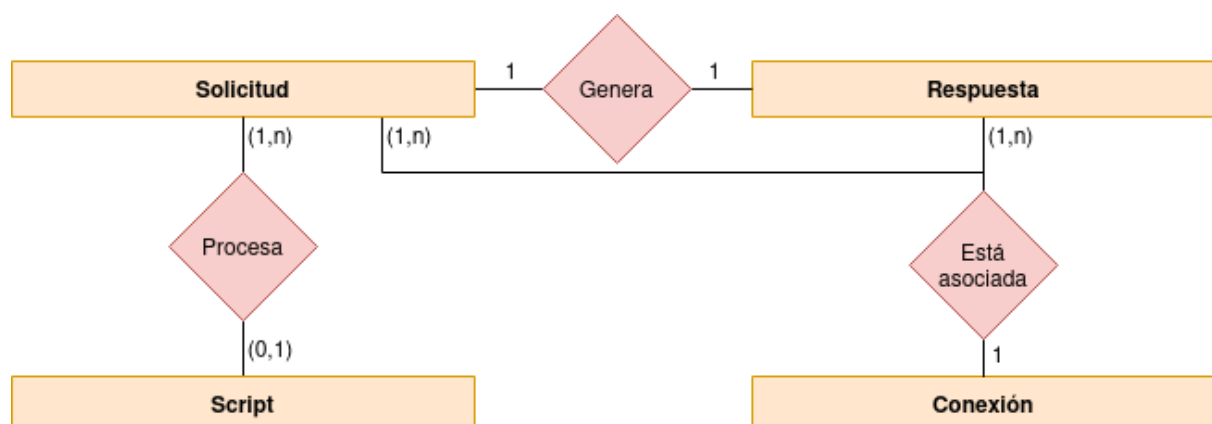
Interfaz de software

El servidor es compatible con sistemas operativos basados en UNIX, incluyendo Linux. En nuestro caso, nuestro entorno de desarrollo se ha realizado en Debian 12.

Interfaz de comunicación

El servidor soporta los protocolos de comunicación web HTTP/1.0 y HTTP/1.1.

Diagrama Entidad-Relación



Casos de uso

Procesamiento de una Solicitud GET

Actores: Cliente.

Precondiciones: El servidor está en funcionamiento y escuchando en el puerto configurado.

Flujo Principal:

1. El cliente envía una solicitud GET al servidor.
2. El servidor recibe la solicitud y analiza los encabezados y la URI.
3. El servidor busca el recurso solicitado.
 - 3.1. Si el recurso es un archivo estático, el servidor prepara una respuesta con el contenido del archivo.
 - 3.2. Si el recurso es un script, el servidor lo ejecuta.
4. El servidor envía la Respuesta al cliente.

Postcondiciones: El cliente recibe el recurso solicitado o un mensaje de error, dependiendo de si se encuentra o hay algún error.

Ejecución de un script mediante una Solicitud POST

Actores: Cliente.

Precondiciones: El servidor está operativo y el recurso solicitado es un script.

Flujo Principal:

1. El cliente envía una solicitud POST, incluyendo datos en el cuerpo de la solicitud.
2. El servidor recibe la solicitud, extrae los datos y ejecuta el script correspondiente, pasando los datos como entrada.
3. El script procesa los datos y genera una salida.
4. El servidor recoge la salida del script, crea una respuesta y la envía de vuelta al cliente.

Postcondiciones: El cliente recibe el resultado de la ejecución del script.

Manejo de Solicitudes Concurrentes

Actores: Múltiples clientes.

Precondiciones: El servidor está configurado para manejar concurrencia.

Flujo Principal:

1. Varios clientes envían solicitudes al servidor simultáneamente.
2. El servidor acepta las conexiones entrantes y procesa cada solicitud en su propio hilo o proceso.
3. Cada solicitud es atendida de manera independiente, y las respuestas son enviadas de vuelta a los respectivos clientes.

Postcondiciones: Todos los clientes reciben sus respuestas en un tiempo adecuado, sin interferencia entre las solicitudes.

Configuración del Servidor Mediante Archivo de Configuración

Actores: Administrador del servidor.

Precondiciones: El servidor está inicialmente configurado con valores predeterminados.

Flujo Principal:

1. El administrador modifica el archivo server.conf para cambiar la configuración.
2. El servidor se reinicia para aplicar los cambios.
3. El servidor opera según la nueva configuración especificada.

Postcondiciones: El servidor refleja la configuración actualizada en su operación.

Respuesta a una Solicitud con Método HTTP no Soportado

Actores: Cliente web.

Precondiciones: El servidor solo soporta GET, POST, y OPTIONS.

Flujo Principal:

1. Cliente envía una solicitud con un método HTTP no soportado (por ejemplo, PUT).
2. El servidor identifica el método como no soportado.
3. El servidor responde con 405 Method Not Allowed.

Postcondiciones: El cliente es informado de que el método de la solicitud no es soportado por el servidor.

Conclusiones personales y técnicas.

El inicio de la práctica fue complicado, ya que no teníamos suficientes conocimientos para desarrollar servidores, clientes y conectarlos, tal y como se solicitaba en el enunciado. Sin embargo, poco a poco fuimos informándonos más a fondo y pudimos conseguir avanzar y salir del estancamiento. Finalmente, se nos echó el tiempo encima y los últimos días previos a la entrega hemos tenido una carga de trabajo bastante elevada, lo cual nos ha perjudicado en el perfeccionamiento de la práctica.

De cara a futuras prácticas, será un imperativo llevar la práctica con constancia y celeridad, con el fin de no acumular el trabajo y poder realizar más pruebas y mejores resultados de rendimientos y fugas de memoria.