

Web Server

Davide Bartoli

24 maggio 2024

Capitolo 1

Descrizione

Il progetto consiste in un semplice web server scritto in Python, utilizzato socket TCP e programmazione concorrente tramite la libreria `threading`. Il progetto permette di visualizzare pagine html tramite un browser.

Capitolo 2

Funzionamento

Il server utilizza il protocollo TCP per ricevere richieste HTTP (solo di tipo GET) e risponde con la pagina html corrispondente o con un errore 404 se la pagina non è stata trovata.

Il server ascolta sulla porta 8080 e viene inizializzato nel seguente modo:

```
SERVER_PORT = 8080
SERVER_ADDRESS = ('', SERVER_PORT)
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
serverSocket.setsockopt(SOL_SOCKET, SO_REUSEPORT, 1)
serverSocket.bind(SERVER_ADDRESS)
```

Per poter gestire più richieste contemporaneamente ogni volta che un cliente si connette viene gestito da un diverso thread:

```
while True:
    print('Ready to serve...')
    connectionSocket, addr = serverSocket.accept()
    print(connectionSocket, addr)
    threading.Thread(target=serve_request, args=(connectionSocket,), daemon=True).start()
```

Da ogni richiesta viene estratto il file voluto dall'utente, se il file non esiste viene ritornato un errore 404:

```
def serve_request(connection):
    try:
        message = connection.recv(1024)
        if len(message.split()) > 0:
            print(message.split()[0], ':', message.split()[1])
            filename = message.split()[1]
            if filename == b'/':
                filename = b'/index.html'
            f = open(filename[1:], 'r+')
            outputdata = f.read()

            connectionSocket.send("HTTP/1.1 200 OK\r\n\r\n".encode())
            connectionSocket.send(outputdata.encode())
            connectionSocket.send("\r\n".encode())
            connectionSocket.close()
    except IOError:
        # File non trovato
        connectionSocket.send(bytes("HTTP/1.1 404 Not Found\r\n\r\n", "UTF-8"))
        connectionSocket.send(bytes(
            "<html><head></head><body><h1>404 Not Found :(</h1></body></html>\r\n", "UTF-8"))
        connectionSocket.close()
```

Capitolo 3

Utilizzo

3.1 Prerequisiti

Per eseguire il programma è necessario avere installato python.

3.2 Utilizzo

Per eseguire il programma basta eseguire il file `server.py`, ad esempio utilizzando il comando `python3 server.py`.

```
➔ server_http python3 server.py
the web server is up on port: 8080
Ready to serve...
█
```

Successivamente è possibile connettersi tramite un browser per vedere le pagine html.

