September 29, 2024

TCP-based Web Application: Single and Multi-Threaded Server with Custom Client

Mehraz Abedin Raz 2022293
Nakul Garg 2022309

Objective & System Overview

Overview

This assignment explores building a TCP-based web application that includes developing a single-request web server, a multi-threaded web server, and a custom HTTP client.

Three Parts:

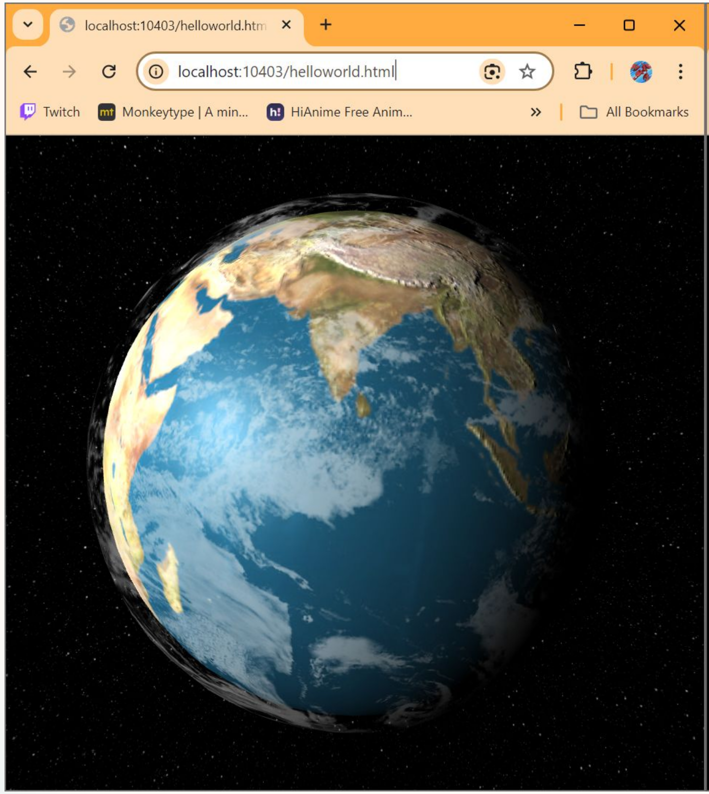
- Single-request server: Handles one HTTP request at a time.
- Multi-threaded server: Manages multiple simultaneous HTTP requests.
- Custom HTTP client: Sends GET requests to the server and displays the response.

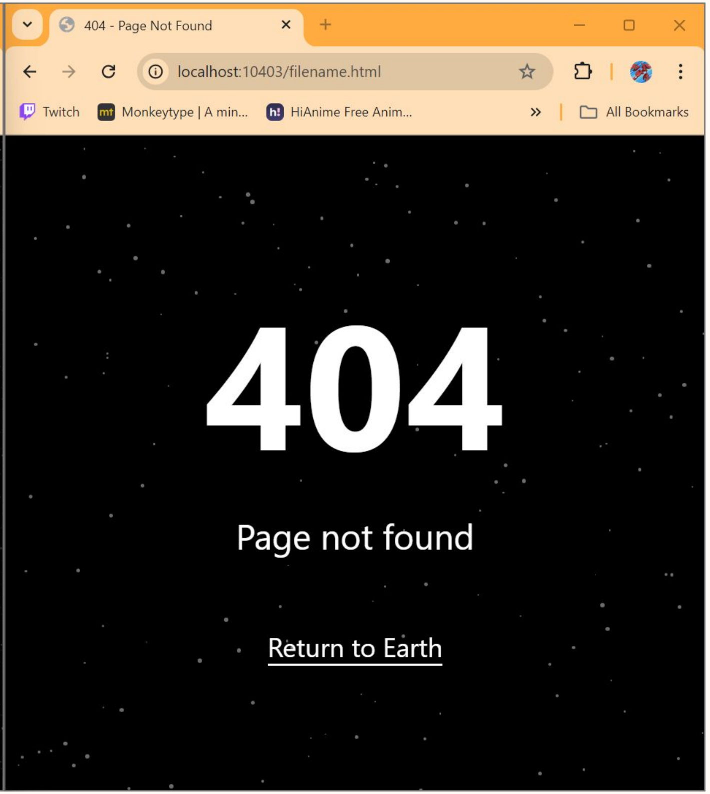
Single-Request Web Server

Key Points:

- Created a TCP socket and bound it to a specific port.
- Server parses incoming HTTP requests and retrieves the requested file.
- Sends HTTP response with the file or a 404 error if the file is not found.

Screenshot - Single request server


localhost:10403/helloworld.htm
localhost:10403/helloworld.html
Twitch
Monkeytype | A min...
HiAnime Free Anim...
All Bookmarks



404

Page not found

Return to Earth

Multi-Threaded Server

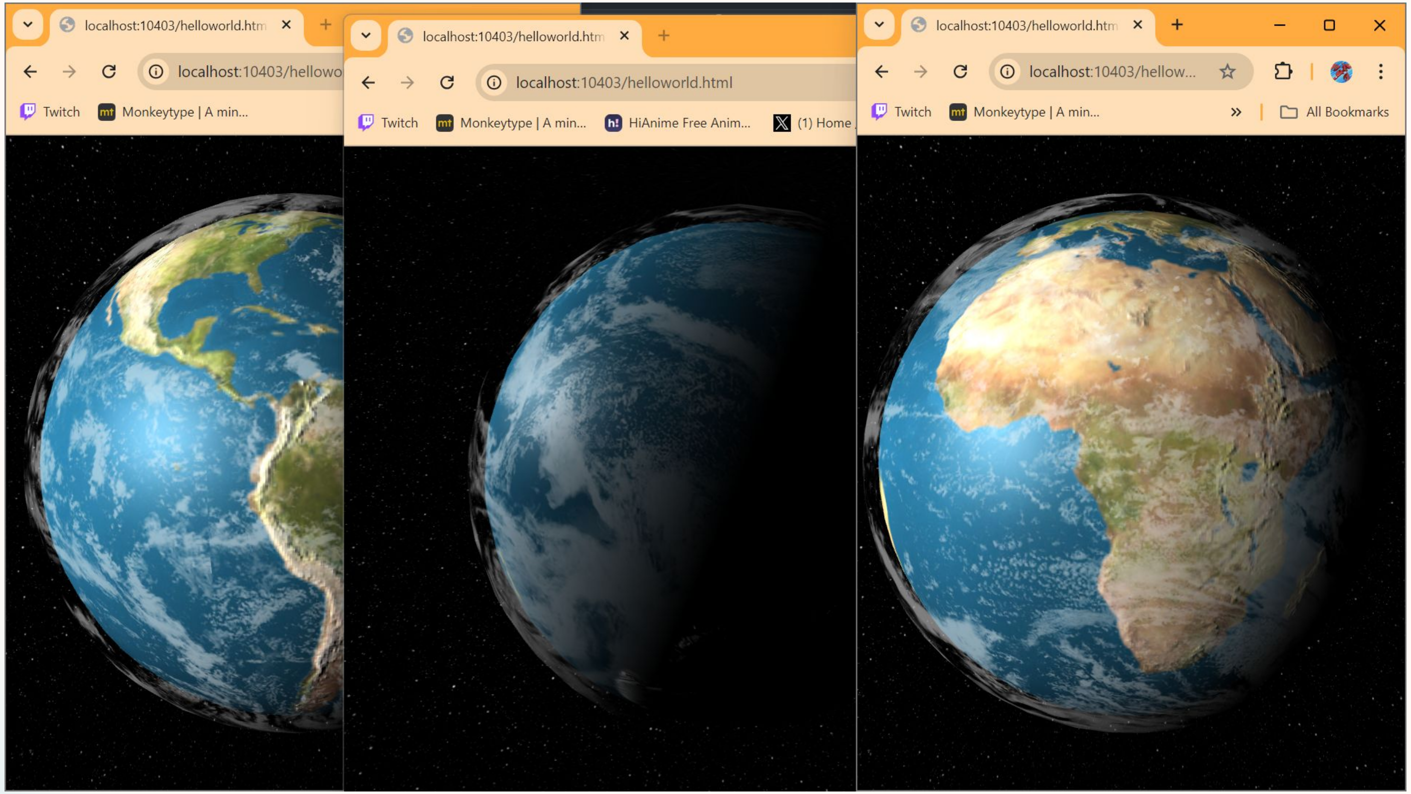
Description

Extended the server to handle multiple concurrent requests using Python threading module. Each new client request is handled in a separate thread.

Key Points

- Main thread listens for connections.
- Separate threads serve each request.

Screenshot - Multi - threaded server


localhost:10403/helloworld.htm
localhost:10403/hellowo
Twitch
Monkeytype | A min...
localhost:10403/helloworld.htm
localhost:10403/helloworld.html
Twitch
Monkeytype | A min...
HiAnime Free Anim...
(1) Home
localhost:10403/helloworld.htm
localhost:10403/hellow...
Twitch
Monkeytype | A min...
All Bookmarks

HTTP Client

Overview

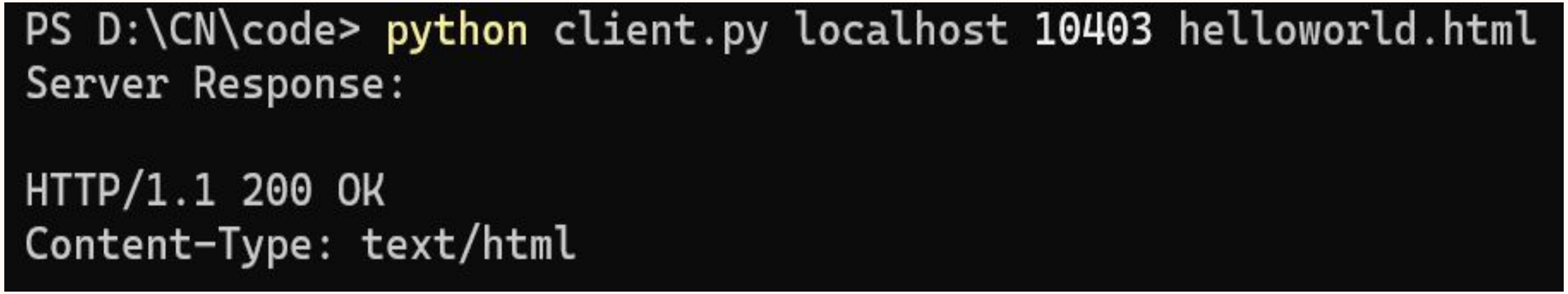
- Created a custom HTTP client that connects to the server, sends HTTP GET requests, and displays the server's responses.
- Command Line Usage Example: `python client.py server_ip server_port filename`
- Screenshot: Terminal output showing a successful client request and the corresponding server response.

404 Not Found Handling

Key Point

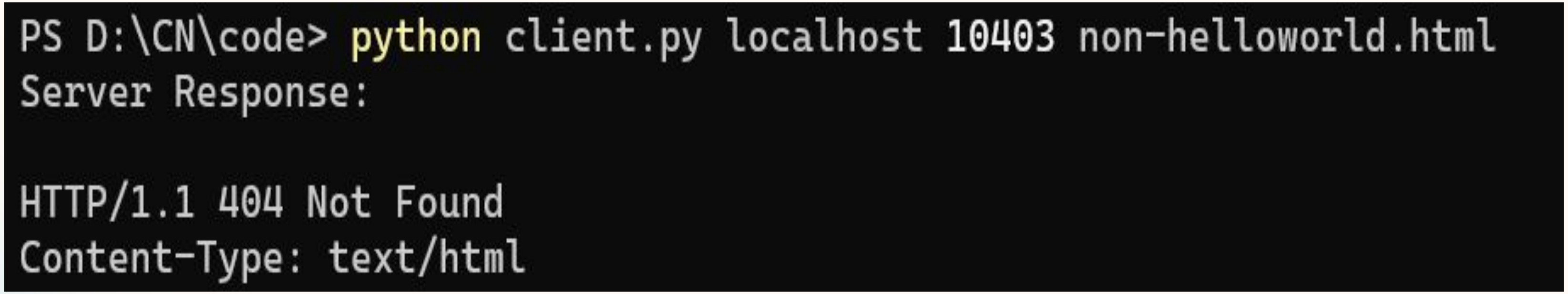
The server sends a "404 Not Found" message when the requested file is missing. This error is handled gracefully to ensure proper communication with the client.

Screenshot - Client response



```
PS D:\CN\code> python client.py localhost 10403 helloworld.html
Server Response:

HTTP/1.1 200 OK
Content-Type: text/html
```



```
PS D:\CN\code> python client.py localhost 10403 non-helloworld.html
Server Response:

HTTP/1.1 404 Not Found
Content-Type: text/html
```

Testing and Results

Summary of Testing:

- Successfully retrieved files and displayed them in the browser or client.
- Handled 404 errors when files were missing.
- Tested multi-client handling to ensure the server could serve multiple requests concurrently.

Testing Methods

Both browser-based and custom client-based testing.

Conclusion

Summary of Skills Learned

- TCP socket programming for client-server applications.
- Parsing and handling HTTP requests.
- Implementing multithreading in Python for concurrent request handling.

Reflection

This assignment emphasized the importance of robust server-client architecture in real-world web applications.

September 29, 2024

Thank you

Mehraz Abedin Raz 2022293
Nakul Garg 2022309