

Documentation: References & Credits

(Academic and Technical Resources Used)

1. Textbooks

(a) Behrouz A. Forouzan, *Data Communication and Networking*, 5th Ed., McGraw Hill, 2017

Used for foundational understanding of:

- HTTP Protocol Fundamentals — request-response model, client-server communication, and layered architecture
- TCP/IP Stack Visualization — relationship between application and lower network layers
- HTTP Methods & Status Codes — semantic roles of GET, POST, PUT, DELETE, and response codes
- Message Structure — HTTP request/response format including headers and status lines

(b) James F. Kurose & Keith W. Ross, *Computer Networking: A Top-Down Approach*, 6th Ed., Pearson, 2017

Referenced for:

- Application Layer Protocols — persistent vs non-persistent connections in HTTP
- Web Caching Concepts — proxy operation and conditional GET requests
- HTTP Headers — roles of Content-Type, Content-Length, User-Agent, and Accept
- Performance Optimization — RTT, pipelining, and parallel connection analysis

(c) William Stallings, *Data and Computer Communication*, 10th Ed., Pearson, 2017

Provided insights on:

- Protocol Design Principles — stateless architecture of HTTP
- Security Considerations — HTTPS, SSL/TLS encryption, and secure data transfer
- Error Handling — correct usage of HTTP error codes

- Data Encoding & Transfer — encoding standards and chunked transfer mechanisms
-

2. Research & Web Documentation

(a) Fielding, R. T. (2000), *Architectural Styles and the Design of Network-based Software Architectures*, UC Irvine

Used for understanding REST architecture, stateless communication, and resource-oriented design principles fundamental to HTTP API structure.

(b) Belshe, M., Peon, R., & Thomson, M. (2015), *Hypertext Transfer Protocol Version 2 (HTTP/2)*, RFC 7540, IETF

Referenced for:

- Evolution from HTTP/1.1 to HTTP/2
- Multiplexing, header compression, and server push mechanisms
- Performance comparison and visualization enhancements

(c) Mozilla Developer Network (MDN), *HTTP Documentation*, 2024

Provided:

- Modern HTTP Standards — updated reference for methods, status codes, and headers
- CORS & Browser Behavior — handling of security and cross-origin policies
- Examples & Use Cases — practical demonstrations in the simulator

(d) Grigorik, I. (2013), *High Performance Browser Networking*, O'Reilly Media

Referenced for:

- Performance Visualization — latency and timing diagrams
- Connection Management — keep-alive, pooling, and performance optimization
- TCP/IP Influence — impact on HTTP responsiveness

(e) Express.js Documentation (2024)

Used for backend functionality:

- Routing & Middleware Design — Express-based server handling

- Request/Response Handling — structured HTTP communication with headers and status codes
-

Development Team

Developed By:

- Sagnik Mitra
- Abhik Sinha

Guided By:

Prof. Swaminathan Annadurai

Department of Computer Science and Engineering
[Vellore Institute of Technology, Chennai]

This project was collaboratively developed as part of an academic initiative to enhance conceptual understanding of web communication through simulation and visualization.

Development Resources

- MDN Web Docs — HTTP standards reference
 - Node.js & Express.js — backend framework for server implementation
 - Axios — HTTP client for practical demonstrations
 - CSS Grid Generators & Design Tools — UI layout and responsive visualization
 - ChatGPT (AI Assistance) — documentation support and conceptual simplification based on cited sources
-

Summary

This documentation acknowledges all academic, technical, and open-source references that guided the development of the HTTP Request-Response Simulator.

Each concept—from protocol architecture and performance optimization to visualization and caching—is grounded in principles drawn from the primary networking textbooks and official web standards listed above.

