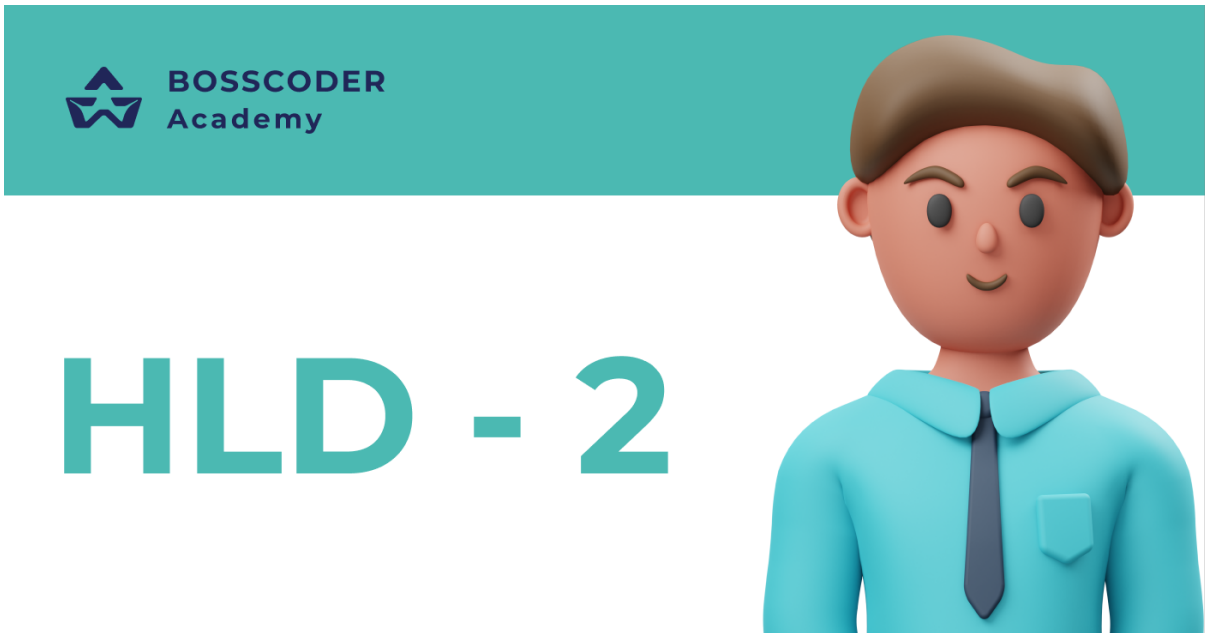
BOSSCODER ACADEMY

Request A Callback



HLD - 2

Abhishek Singh
2nd August, 2023

Network Protocols:

Network protocols are a set of rules and conventions that govern communication between devices and systems within a network. They define how data is transmitted, routed, and received over a network, ensuring seamless and efficient communication. Examples of network protocols include IP (Internet Protocol), TCP (Transmission Control Protocol), UDP

IP (Internet Protocol):

The IP (Internet Protocol) is a fundamental network protocol used for identifying and locating devices on the internet or any network that follows the TCP/IP suite. IP addresses serve as unique identifiers for devices, enabling data packets to be routed from source to destination across the network.

IP Packet:

An IP packet is a unit of data sent over an IP network. It contains the payload (actual data) and headers with information about the packet, such as source and destination IP addresses, time-to-live (TTL), and checksum. IP packets are the building blocks of data transmission in IP-based networks.

TCP (Transmission Control Protocol):

TCP is a reliable, connection-oriented protocol used for data transmission over the Internet and local networks. It ensures that data is delivered accurately and in sequence by establishing a connection between the sender and receiver before data exchange. TCP uses acknowledgements and flow control mechanisms to handle data loss and congestion.

Request-Response Paradigm:

The request-response paradigm is a communication pattern where a client sends a request to a server, and the server responds with the requested information or action. This model is widely used in web applications and APIs, where clients (e.g., web browsers or mobile apps) send HTTP requests

HTTP (Hypertext Transfer Protocol):

HTTP is an application layer protocol used for transmitting hypermedia documents, such as HTML web pages, over the internet. It follows the request-response paradigm, where clients send HTTP requests to servers, and servers respond with the requested resources. HTTP is the foundation of web communication and is widely used for web browsing and data exchange.

HTTPS (Hypertext Transfer Protocol Secure):

HTTPS is a secure version of HTTP that adds an extra layer of security using SSL/TLS encryption. It ensures that data exchanged between the client and server is encrypted, protecting it from eavesdropping and unauthorized access. HTTPS is commonly used for secure transactions, such as online banking and e-commerce.

Storage Database:

A storage database is a structured collection of data that is organized and stored for easy retrieval and manipulation. It is an essential component of many applications, providing a persistent and reliable way to store and manage data.

Persistent and Non-Persistent Database:

A persistent database stores data permanently, even after the application or system is shut down. It retains data across sessions and allows users to access it at a later time. In contrast, a non-persistent database stores data

Latency and Throughput:

Latency refers to the time delay between sending a request and receiving a response. It is the time taken for data to travel from the sender to the receiver. Throughput, on the other hand, is the rate at which data is transmitted or processed. It represents the amount of data that can be transmitted or processed per unit of time.

Caching - Reduce Latency:

Caching is a technique used to reduce latency and improve system performance by storing frequently accessed data in a cache. A cache is a high-speed memory that stores copies of frequently used data or resources. When a client requests data, the system first checks the cache. If the data is found in the cache, it is retrieved quickly, reducing the need to fetch it from the original source.

a. Write-Through Cache: In a write-through cache, data is written simultaneously to both the cache and the original data store. This ensures that the cache always contains the most up-to-date data. However, write-through caching can lead to increased latency for write operations as each write must be propagated to the underlying data store.

b. Write-Back Cache: In a write-back cache, data is first written to the cache, and the update is deferred for writing to the original data store. This approach improves write performance as data is written to the cache quickly, and the actual update to the data store can be done later during periods of lower activity.

Cache eviction is the process of removing items from the cache to make room for new or more frequently accessed data. Popular cache eviction techniques include Least Recently Used (LRU), First-In-First-Out (FIFO), and Random replacement. These techniques determine which items to evict based on their access patterns and frequency.

Stateless and Stateful Application:

A stateless application does not retain any client-specific information between requests. Each request from the client is treated independently, and the server does not maintain any session information. In contrast, a stateful application retains client-specific information between requests, allowing it to maintain the state of the session and provide personalized services to the client.

Assessment: https://www.bosscoderacademy.com/blog/hld-2-assessment