ONE PAGE SUMMARY

IT & Software › Backend Development › WebSockets

Introduction to WebSockets

WebSockets enable real-time communication in web applications.

DEFINITION

> WebSockets are a protocol that allows for full-duplex communication between a client and server over a single, long-lived connection. This enables real-time data exchange without the need for constant HTTP requests.

KEY CONCEPTS

Full-Duplex Communication

Both the client and server can send and receive messages simultaneously, improving interaction speed.

Persistent Connection

WebSockets maintain an open connection, reducing latency and overhead compared to traditional HTTP requests.

Event-Driven Model

WebSockets use events to trigger actions, allowing for responsive applications that react to user inputs instantly.

Binary and Text Data

WebSockets can transmit both binary and text data, making them versatile for various applications.

Cross-Origin Communication

WebSockets can facilitate communication between different domains, enhancing the capabilities of web applications.

EXAMPLES

→ Chat applications using WebSockets for real-time messaging.

→ Live sports updates delivered instantly to users via WebSocket connections.

→ Collaborative document editing where changes appear in real-time.

MEMORY TIPS

★ Think of WebSockets as a 'phone call' where both parties can talk at once.

★ Remember 'open line' for persistent connection – it stays connected until closed.

★ Use 'event' to recall the event-driven model; actions happen based on user events.

COMMON MISTAKES

✗ Assuming WebSockets are the same as HTTP; they serve different purposes.

✗ Neglecting to handle connection closure properly, leading to resource leaks.

✗ Overcomplicating data formats; keep messages simple for efficiency.

QUICK RECAP

WebSockets provide a way for real-time, two-way communication between clients and servers. They maintain a persistent connection, allowing for efficient data transfer and immediate updates in applications.