

**MAKALAH REFLEKSI PEMBELAJARAN
PEMROGRAMAN JARINGAN**



Disusun Oleh:

FITRI

231401003

Dosen Pengampu: Ucok,S.KOM.,MT

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS INDONESIA TIMUR
MAKASSAR**

2026

I. PENDAHULUAN

Pemrograman jaringan adalah cabang ilmu komputer yang mempelajari cara melakukan komunikasi antar komputer melalui jaringan. Dalam era digital yang semakin maju, aplikasi jaringan menjadi bagian penting dari kehidupan sehari-hari: mulai dari web browser, layanan REST API, komunikasi real-time, hingga sistem Internet of Things (IoT). Buku Ajar Pemrograman Jaringan membekali pembaca dengan teori dan praktik dari tingkat dasar sampai lanjutan untuk membangun aplikasi jaringan secara sistematis dan komprehensif.

II. TUJUAN PEMBELAJARAN

Setelah mempelajari buku ajar ini, mahasiswa diharapkan mampu:

1. Memahami dasar-dasar pemrograman jaringan dan arsitektur jaringan komputer.
2. Mengimplementasikan protokol komunikasi seperti TCP, UDP, HTTP, dan MQTT dalam aplikasi nyata.
3. Membangun layanan web dan REST API menggunakan Python.
4. Menerapkan komunikasi real-time melalui WebSocket.
5. Menerapkan teknik keamanan jaringan seperti SSL/TLS dalam aplikasi client-server.
6. Mengembangkan aplikasi IoT sederhana menggunakan protokol MQTT.
7. Mengintegrasikan seluruh konsep tersebut dalam proyek akhir sistem terdistribusi.

III. PEMBAHASAN

1. Konsep Dasar Pemrograman Jaringan

Pemrograman jaringan melibatkan komunikasi antar perangkat yang terhubung melalui protokol standar seperti TCP/IP. Konsep dasar mencakup pemahaman socket, alamat IP, port, serta model client-server. Socket API memungkinkan aplikasi membuka koneksi dan bertukar data dengan peer di jaringan.

2. Protokol TCP dan UDP

TCP (Transmission Control Protocol) memberikan koneksi yang andal dan teratur, cocok untuk aplikasi seperti chat dan web server.

UDP (User Datagram Protocol) adalah protokol tanpa koneksi yang lebih cepat dan ringan, cocok untuk streaming data real-time.

3. Concurrency dan Async I/O

Untuk menangani banyak koneksi sekaligus, buku ini membahas teknik concurrency:

- Threading dan Multiplexing untuk mengelola banyak socket.
- Asynchronous I/O (asyncio) untuk menangani operasi input/output tanpa blocking.

4. HTTP & REST API

HTTP merupakan protokol request-response yang banyak digunakan untuk layanan web. REST API menyediakan cara standar aplikasi berbicara dengan server melalui metode HTTP seperti GET, POST, PUT, dan DELETE. Pembelajaran ini ditunjang pustaka Python 'requests' untuk mengonsumsi layanan eksternal.

5. WebSocket

WebSocket memperkenalkan komunikasi full duplex yang memungkinkan server dan client bertukar data secara real-time di atas satu koneksi TCP yang persisten — ideal untuk aplikasi seperti dashboard saham atau chat real-time.

6. Keamanan Jaringan (SSL/TLS)

Keamanan komunikasi diperlukan untuk melindungi data dari penyadapan atau serangan man-in-the-middle. SSL/TLS menerapkan kriptografi modern untuk mengenkripsi data, sehingga aplikasi client-server dapat saling bertukar informasi secara aman.

7. IoT dan MQTT

MQTT adalah protokol ringan untuk IoT yang menggunakan publish/subscribe pada broker pesan. Ini memungkinkan banyak sensor dan aplikasi berkomunikasi secara efisien, bahkan pada jaringan dengan sumber daya terbatas.

8. Capstone Project

Sebagai puncak pembelajaran, mahasiswa diwajibkan membangun sistem terdistribusi yang mengintegrasikan beberapa protokol (misalnya MQTT, HTTP, WebSocket) dalam satu aplikasi utuh — mencakup agent, server inti, dan antarmuka monitor.

IV. KESIMPULAN

Buku Ajar Pemrograman Jaringan memberikan pembelajaran terstruktur dari dasar hingga lanjutan untuk memahami dan mengimplementasikan konsep pemrograman jaringan modern. Dengan materi yang mencakup protokol TCP/UDP, HTTP/REST, WebSocket, keamanan SSL, hingga IoT dengan MQTT, buku ini mempersiapkan mahasiswa untuk mengembangkan aplikasi jaringan nyata serta menghadapi tantangan teknologi terkini.

V. SARAN

Untuk maksud pembelajaran yang lebih efektif, disarankan mahasiswa:

1. Mencoba semua contoh kode praktikum secara langsung.
2. Memahami pola desain jaringan dan pemilihan protokol sesuai kebutuhan aplikasi.
3. Mengembangkan proyek akhir dengan dokumentasi lengkap.
4. Menguji keamanan dan reliability aplikasi dalam berbagai kondisi.