

**MAKALAH REFLEKSI PEMBELAJARAN  
PEMOGRAMAN JARINGAN**



**DISUSUN OLEH:**

**IIN SUHANA (231401002)**

**MATA KULIAH: PEMOGRAMAN JARINGAN**

**DOSEN PENGAMPU: Ucok,.S.Kom.,MT**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS INDONESIA TIMUR  
2026**

## KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan makalah yang berjudul "**Pemrograman Jaringan**" dengan baik. Makalah ini disusun untuk memenuhi tugas mata kuliah Pemrograman Jaringan serta sebagai bahan pembelajaran bagi mahasiswa dalam memahami konsep dasar hingga penerapan pemrograman jaringan.

Makalah ini membahas berbagai materi penting, mulai dari konsep dasar pemrograman jaringan, socket API, protokol TCP dan UDP, teknik concurrency, serialisasi data, hingga teknologi jaringan modern seperti HTTP, REST API, WebSocket, keamanan jaringan, serta sistem terdistribusi dan Internet of Things (IoT).

Penulis menyadari bahwa makalah ini masih memiliki keterbatasan dan kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi penyempurnaan makalah ini di masa mendatang. Semoga makalah ini dapat memberikan manfaat dan menambah wawasan bagi pembaca.

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Perkembangan teknologi informasi dan komunikasi saat ini mengalami kemajuan yang sangat pesat. Jaringan komputer menjadi salah satu komponen utama yang mendukung pertukaran informasi secara cepat dan efisien. Hampir seluruh aplikasi modern seperti website, aplikasi mobile, media sosial, layanan streaming, dan Internet of Things (IoT) memanfaatkan jaringan komputer sebagai media komunikasi data.

Pemrograman jaringan merupakan teknik pemrograman yang digunakan untuk membangun aplikasi yang dapat saling berkomunikasi melalui jaringan. Dengan pemrograman jaringan, sebuah aplikasi dapat mengirim dan menerima data antar perangkat, baik dalam jaringan lokal maupun melalui internet. Oleh karena itu, pemahaman tentang pemrograman jaringan menjadi sangat penting, khususnya bagi mahasiswa di bidang teknologi informasi.

Dengan menguasai pemrograman jaringan, mahasiswa diharapkan mampu memahami cara kerja komunikasi data serta dapat mengembangkan aplikasi jaringan yang efisien, aman, dan dapat digunakan oleh banyak pengguna secara bersamaan.

#### **1.2 Rumusan Masalah**

Berdasarkan latar belakang tersebut, maka rumusan masalah dalam makalah ini adalah sebagai berikut:

1. Apa yang dimaksud dengan pemrograman jaringan?
2. Apa saja komponen utama dalam pemrograman jaringan?
3. Bagaimana peran socket dalam komunikasi jaringan?
4. Apa perbedaan dan penggunaan protokol TCP dan UDP?

#### **1.3 Tujuan Penulisan**

Tujuan penulisan makalah ini adalah untuk menjelaskan konsep dasar pemrograman jaringan, memahami cara kerja socket dan protokol jaringan, serta mengetahui penerapan pemrograman jaringan dalam berbagai aplikasi modern.

#### **1.4 Manfaat Penulisan**

Manfaat dari penulisan makalah ini adalah untuk menambah wawasan dan pengetahuan pembaca mengenai pemrograman jaringan, serta sebagai referensi pembelajaran bagi mahasiswa dalam memahami dan mengembangkan aplikasi berbasis jaringan.

## **BAB II**

### **KONSEP DASAR PEMROGRAMAN JARINGAN**

#### **2.1 Pengertian Pemrograman Jaringan**

Pemrograman jaringan adalah proses pembuatan aplikasi yang memungkinkan komputer atau perangkat lain saling berkomunikasi melalui jaringan. Komunikasi ini dilakukan dengan mengirim dan menerima data menggunakan protokol tertentu yang telah ditetapkan.

Pemrograman jaringan banyak digunakan pada aplikasi seperti email, aplikasi chat, web server, game online, dan sistem IoT. Dengan pemrograman jaringan, aplikasi dapat bekerja secara terdistribusi dan melayani banyak pengguna secara bersamaan.

#### **2.2 Komponen Pemrograman Jaringan**

Komponen utama dalam pemrograman jaringan meliputi:

- **Client:** aplikasi atau perangkat yang mengirim permintaan.
- **Server:** aplikasi atau perangkat yang menerima dan memproses permintaan.
- **Alamat IP,** digunakan untuk mengidentifikasi perangkat dalam jaringan.
- **Port,** digunakan sebagai jalur komunikasi pada suatu perangkat.
- **Protokol,** yaitu aturan komunikasi data seperti TCP dan UDP.

#### **2.3 Model Client–Server**

Model client–server adalah model komunikasi jaringan yang paling umum digunakan. Client bertugas mengirim permintaan data, sedangkan server bertugas memberikan respon sesuai permintaan tersebut. Model ini banyak digunakan pada sistem web dan aplikasi jaringan lainnya.

## **BAB III**

### **SOCKET API DASAR**

#### **3.1 Pengertian Socket**

Socket adalah titik akhir komunikasi yang digunakan oleh aplikasi untuk mengirim dan menerima data melalui jaringan. Socket menjadi penghubung antara aplikasi dan protokol jaringan.

#### **3.2 Jenis-Jenis Socket**

Jenis socket yang umum digunakan adalah:

- **Socket TCP**, digunakan untuk komunikasi yang andal.
- **Socket UDP**, digunakan untuk komunikasi yang cepat tanpa jaminan keandalan.

#### **3.3 Cara Kerja Socket API**

Cara kerja socket API meliputi beberapa tahap, yaitu pembuatan socket, pengikatan alamat dan port, pembuatan koneksi, pengiriman dan penerimaan data, serta penutupan koneksi.

## **BAB IV**

### **PROTOKOL TCP (APLIKASI CHAT)**

#### **4.1 Pengertian Protokol TCP**

TCP (Transmission Control Protocol) adalah protokol komunikasi yang bersifat andal dan berorientasi koneksi. TCP memastikan data sampai ke tujuan tanpa kehilangan dan dalam urutan yang benar.

#### **4.2 Karakteristik TCP**

Karakteristik utama TCP antara lain:

- Berorientasi koneksi
- Menjamin keandalan data
- Mengatur urutan pengiriman data

#### **4.3 Implementasi TCP pada Aplikasi Chat**

Pada aplikasi chat, TCP digunakan untuk memastikan pesan yang dikirim oleh pengguna dapat diterima secara lengkap oleh penerima. TCP sangat cocok digunakan karena pesan chat tidak boleh hilang atau tertukar.

## **BAB V**

### **PROTOKOL UDP (STREAMING & BROADCASTING)**

#### **5.1 Pengertian UDP**

UDP (User Datagram Protocol) adalah protokol komunikasi jaringan yang tidak berorientasi koneksi. Artinya, UDP tidak membangun koneksi terlebih dahulu sebelum mengirim data. Data dikirim dalam bentuk paket tanpa jaminan apakah paket tersebut sampai ke tujuan atau tidak.

UDP lebih mengutamakan kecepatan dibandingkan keandalan. Oleh karena itu, protokol ini cocok digunakan pada aplikasi yang membutuhkan pengiriman data secara cepat.

#### **5.2 Perbedaan TCP dan UDP**

Perbedaan utama antara TCP dan UDP terletak pada keandalan dan cara pengiriman data. TCP menjamin data sampai dengan urutan yang benar, sedangkan UDP tidak menjamin keutuhan data. Namun, UDP memiliki keunggulan dalam hal kecepatan pengiriman.

#### **5.3 Penerapan UDP pada Streaming dan Broadcasting**

UDP banyak digunakan pada aplikasi streaming video, audio, dan broadcasting. Pada aplikasi ini, kehilangan sebagian kecil data tidak terlalu berpengaruh dibandingkan keterlambatan pengiriman data.

## **BAB VI**

### **ERROR HANDLING & FRAMING DATA**

#### **6.1 Pengertian Error Handling**

Error handling adalah proses penanganan kesalahan yang terjadi selama komunikasi jaringan. Kesalahan dapat terjadi karena gangguan jaringan, koneksi terputus, atau data yang tidak sesuai format.

#### **6.2 Jenis-Jenis Error**

Beberapa jenis error dalam pemrograman jaringan antara lain timeout, koneksi terputus, dan kesalahan format data.

#### **6.3 Framing Data dalam Komunikasi Jaringan**

Framing data digunakan untuk menentukan batas awal dan akhir suatu pesan. Dengan framing yang baik, data dapat diterima dan diproses dengan benar oleh penerima.

## **BAB VII**

### **CONCURRENCY PART I (THREADING)**

#### **7.1 Pengertian Concurrency**

Concurrency adalah kemampuan suatu sistem untuk menangani beberapa proses atau koneksi secara bersamaan. Dalam pemrograman jaringan, concurrency sangat penting agar server dapat melayani banyak client.

#### **7.2 Konsep Threading**

Threading adalah teknik concurrency yang membagi proses menjadi beberapa thread. Setiap thread dapat menangani satu client.

#### **7.3 Implementasi Threading**

Dengan threading, server dapat tetap berjalan dengan baik meskipun melayani banyak client secara bersamaan.

## **BAB VIII**

### **SERIALISASI DATA (JSON & PICKLE)**

#### **8.1 Pengertian Serialisasi Data**

Serialisasi data adalah proses mengubah data menjadi format tertentu agar dapat dikirim melalui jaringan atau disimpan.

#### **8.2 JSON**

JSON adalah format data yang ringan, mudah dibaca, dan banyak digunakan dalam komunikasi jaringan.

#### **8.3 Pickle**

Pickle adalah metode serialisasi khusus Python yang digunakan untuk menyimpan objek Python.

#### **8.4 Perbandingan JSON dan Pickle**

JSON lebih aman dan universal, sedangkan Pickle lebih fleksibel tetapi kurang aman.

## **BAB IX**

### **ASYNCHRONOUS I/O (CONCURRENCY PART II)**

#### **9.1 Pengertian Asynchronous I/O**

Asynchronous I/O memungkinkan program melakukan proses input dan output tanpa harus menunggu proses lain selesai.

#### **9.2 Kelebihan Asynchronous**

Teknik ini lebih efisien dan hemat sumber daya.

#### **9.3 Penerapan Asynchronous I/O**

Banyak digunakan pada server modern dengan banyak koneksi client.

## **BAB X**

### **I/O MULTIPLEXING (SELECT & POLL)**

#### **10.1 Pengertian Select**

Select digunakan untuk memantau banyak socket dalam satu waktu.

#### **10.2 Pengertian Poll**

Poll memiliki fungsi serupa dengan select namun lebih fleksibel.

#### **10.3 Perbandingan Select dan Poll**

Poll lebih cocok digunakan pada aplikasi dengan banyak koneksi.

## **BAB XI**

### **PROTOKOL HTTP & WEB SERVER**

#### **11.1 Pengertian HTTP**

HTTP adalah protokol komunikasi yang digunakan pada web.

#### **11.2 Cara Kerja Web Server**

Web server menerima permintaan dari client dan mengirimkan respon.

#### **11.3 Implementasi Web Server Sederhana**

Web server sederhana dapat dibuat menggunakan socket atau bahasa pemrograman seperti Python.

## **BAB XII**

### **REST API & WEB SERVICES**

#### **12.1 Pengertian REST API**

REST API adalah layanan web berbasis HTTP untuk pertukaran data.

#### **12.2 Metode HTTP**

Metode HTTP meliputi GET, POST, PUT, dan DELETE.

#### **12.3 Contoh Penggunaan REST API**

REST API banyak digunakan pada aplikasi web dan mobile.

## **BAB XIII**

### **REAL TIME COMMUNICATION (WEB SOCKET)**

#### **13.1 Pengertian WebSocket**

WebSocket memungkinkan komunikasi dua arah secara real-time.

#### **13.2 Cara Kerja WebSocket**

Koneksi tetap terbuka antara client dan server.

#### **13.3 Penerapan Komunikasi Real-Time**

Digunakan pada aplikasi chat dan monitoring data.

## **BAB XIV**

### **KEAMANAN JARINGAN (NETWORK SECURITY)**

#### **14.1 Pengertian Keamanan Jaringan**

Keamanan jaringan bertujuan melindungi data dan sistem.

#### **14.2 Jenis Ancaman Keamanan**

Ancaman meliputi malware, hacking, dan pencurian data.

#### **14.3 Upaya Pengamanan Jaringan**

Pengamanan dilakukan dengan enkripsi, firewall, dan autentikasi.

## **BAB XV**

### **PENUTUP**

#### **15.1 Kesimpulan**

Pemrograman jaringan merupakan bagian penting dalam pengembangan aplikasi modern. Dengan memahami protokol, socket, dan teknik concurrency, aplikasi jaringan dapat dibangun dengan lebih efisien dan aman.

#### **15.2 Saran**

Mahasiswa disarankan untuk banyak melakukan praktik agar pemahaman pemrograman jaringan semakin baik.