

LAPORAN PRAKTIKUM

953633205  
  
PRAKTIKUM SISTEM TERDISTRIBUSI

MODUL: 3

“PEMROGRAMAN SOCKET DENGAN KONSEP”

“MULTI-THREADING”

Eska Smara Nofiansi M0516018

Fabian Gilanggi M0517014

Henny Nurcahyaning Tyas M0517022

KELOMPOK : 6

HARI : JUMAT

TANGGAL : 17 APRIL 2020

WAKTU : 12.45 – 15.00 WIB

ASISTEN : HAIDAR HENDRI SETYAWAN M0516023

TAUFIQ ODHI DWI PUTRA M0516043

PROGRAM STUDI INFORMATIKA

UNIVERSITAS SEBELAS MARET

2020

Modul 2

PEMROGRAMAN SOCKET DENGAN KONSEP MULTI-THREADING

Eska Smara Nofiansi M0516018, Fabian Gilanggi M0517014, Henny Nurcahyaning Tyas M0517022/ Kelompok 6/ Jumat, 17 April 2020

[fgilanggi@student.uns.ac.id,](mailto:fgilanggi@student.uns.ac.id,) [hnytyas72@student.uns.ac.id,](mailto:hnytyas72@student.uns.ac.id,) [rarasmara@student.uns.ac.id](mailto:rarasmara@student.uns.ac.id)

Asisten : Haidar Hendri Setyawan

**Taufiq Odhi Dwi Putra**

***Abstraksi – Pengertian socket adalah interface pada jaringan yang menjadi titik komunikasi antarmesin pada Internet Protocol. Jika tidak terjadi komunikasi maka tidak ada pertukaran data dan informasi jaringan. Oleh karena itu dilakukan pemrograman untuk menggunakan mengatur lalu lintas data pada socket tersebut. Pada percobaan kali ini menggunakan konsep Multi-threading, dimana koneksi ke server akan diciptakan suatu thread baru.***

***Kata Kunci - Socket, Multi-Threading, Komunikasi Client-Server.***

# Pendahuluan

Socket adalah sebuah abstaksi perangkat lunak yang digunakan suatu “terminal” dari suatu hubugan antara dua mesin atau proses yang saling berinterkoneksi. Socket dapat melakukan beberapa operasi yang meliputi koneksi ke main remote, mengirim data (Write), menerima data (Read), menutup koneksi (Close), bind to port, listen pada data yang masuk dan menerima koneksi dari mesin remote pada port tertentu.

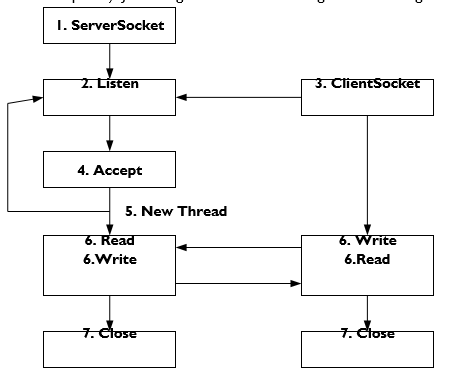
Multi-Threading adalah sebuah mekanisme dimana dalam suatu proses ada beberapa thread yang mengerjakan tugasnya masing-masing dalam waktu yang bersamaan. Dengan Multi-threading proses dapat dilakukan lebih cepat, dimana thread ini bekerja secara pararel.

Dalam laporan ini kami membuat socket programming client server TCP dan UDP dengan konsep multi-threading untuk proses penjumlahan. Socket server yang didalamnya terdapat sustu method untuk melakukan penjumlahan suatu data array bertipe integer, dengan mengambil indeks maksimal 5x. Pada socket client mengirimkan data yang akan dijumlahkan, dan socket server melakukan proses penjumlahan dan mengirimkan hasilnya ke socket lain.

# Dasar Teori

* 1. *Multi-Threading*

Penggunaan Multi-Threading untuk pemrogaman socket sangat penting, karena suatu server harus melayani banyak client. Aplikasi socket untuk server hanya dapat melayani client tunggal, setelah itu server berhenti. Dengan menggunakan konsep multi-threading maka setiap client koneksi ke server akan diciptakan thread baru. Sehingga server selalu running dan siap menerima request dari client (seperti program web server apache).



Gambar 1. Diagram block

* 1. *Java Thread*

Ketika pembuatan object baru class *java.lang.Thread* memberikan penanganan control dan sinkronisasi eksekusinya. Sebuah object yang diperlukan sebuah thread harus menerapkan interface *java.lang.runnable*. Dalam interface ini menyediakan suatu method kunci run().

*public interface Runnable{*

*Abstract public void run();*

*}*

Sebuah thread dimulai siklus hidupnya dengan memanggil method run(). Untuk membuat sebuah class yang akan bertindak sebagai sebuah thread, didefinisikan dengan bentuk sebagai berikut :

*Class namaClass implements Runnable {*

*Void run(){*

*//implementasi kode.....*

*}*

*}*

* 1. *Pengontrolan Thread*

Method *start()* digunakan untuk memulai sebuah thread dieksekusi. Method *stop()* digunakan untuk menghentikan eksekusi suatu thread. Method *suspend()* digunakan untuk menghentikan sementara waktu (pause) untuk kemudian dapat dilanjutkan kembali dengan menggunakan method *resume()*. Jika beban yang harus dikerjakan oleh thread berat/mahal, maka dapat digunakan method *suspend()* dan *resume()*. Method static *sleep(n\_msec)* digunakan untuk menghentikan thread yang active selama n millisecond.

# ALAT DAN LANGKAH PERCOBAAN

Alat yang digunakan adalah sebagai berikut:

1. Laptop
2. IDE Java Eclipse

Langkah dalam melakukan pemrograman socket dengan konsep Multi-Threading antara lain

1. Pembuatan Object baru
2. Membuat instance object dari class (implementasi interface runnable)
3. Buat object class java.lang. Thread dengan parameter konstructor
4. Panggil Method Start
5. Menciptakan sebuah output data stream
6. Pengontrolan Thread, start(), stop(), suspend(), resume(), dll.
7. Close socket

# Hasil dan Analisis Percobaan

Program terdiri dari 2 program yaitu UTP dan UDP

1. UTP

2. UDP

# Kesimpulan

Dalam model paket UDP, komunikasi yang berlangsung harus dengan kesepakatan antara clint dengan server. Sehingga paket yang dikirim akan benar-benar sampai pada penerima dan sesuai dengan kriteria yang dibutuhkan penerima. Hal ini dikarenakan jika terjadi masalah maka pengirim akan melakukan retransmisi. Pada percobaan yang dilakukan, server berhasil menerima input angka yang diinginkan user untuk dijumlahkan. Server juga berhasil memberikan hasil penjumlahan tersebut sehingga client dapat menampilkannya ke layar.

# DaftarPustaka

Winarno. 2020. *Modul Praktikum Sistem Terdistribusi”*.

Oracle. 2017. “*All About Sockets”*. *<https://docs.oracle.com/javase/tutorial/networking/sockets/>*

**[Eska Smara Nofiansi]** Mahasiswa Informatika     Angkatan 2016

**[Fabian Gilanggi]** Mahasiswa Informatika         Angkatan 2017

**[Henny Nurcahyaning Tyas]** Mahasiswa Informatika     Angkatan 2017