

LAPORAN PRAKTIKUM

953633205  
  
PRAKTIKUM SISTEM TERDISTRIBUSI

MODUL: 1

“HTTP (HyperText Transport Protocol)

(Menggunakan Pemrograman Java)”

Albi Revlin Bagaskara M0516008

Alim Wicaksono H. P. M0516009

Eska Smara Nofiansi M0516018

Zulfikar Eka Sagata M0516046

KELOMPOK : 5

HARI : SELASA

TANGGAL : 2 April 2019

WAKTU : 11.10

ASISTEN : PINDYA KHABIIBAH

PROGRAM STUDI INFORMATIKA

UNIVERSITAS SEBELAS MARET

2019

Modul 1

HTTP (*HyperText Transport Protocol*)

Menggunakan Pemrograman Java

Albi Revlin Bagaskara M0516008, Alim Wicaksono H. P. M0516009, Eska Smara Nofiansi M0516018, Zulfikar Eka Sagata M0516046, Kelompok 5 / Selasa, 2 April 2019

Asisten : Pindya Khabiibah

***Abstraksi – Pengertian HTTP merupakan hypertext transport protocol (RFC 1945) adalah protocol yang digunakan untuk komunikasi antara web browser dengan web server. Pada percobaan ini akan dibangun aplikasi sederhana menggunakan HTTP sederhana yaitu HTTPServer yang akan menerima HTTPRequest dan mengirimkan HTTPResponse ke HTTPClient. Nantinya, client hanya akan menggunakan software browser.***

***Kata Kunci – HTML, Response, Request, Client, Server.***

# Pendahuluan

Http (*hypertext transport protocol)* (RFC 1945) adalah protocol yang digunakan untuk komunikasi antara web browser dengan web server. Setelah server dalam keadaan running nanti, maka dapat membuka browser kemudian ketik URL : *http://localhost:8080/index.php.* Maka pada browser akan menampilkan file index.php tersebut layaknya mengakses web server apache.

# Dasar Teori

* 1. *HTTP*

HTTP merupakan implementasi dari protocol TCP, dan menggunakan port 80 sebagai defaultnya. HTTP bersifat stateless, artinya tidak ada informasi yang disimpan, selain itu juga bersifat request-response. HTTP client (user agent misalnya) mengirimkan permintaan (request) ke HTTP server dan server meresponse sesuai dengan permintaan/request HTTP client. User agent software yang bertindak sebagai HTTP client (yang melakukan request) contohnya seperti browser Explorer atau software browser yang berbasis text seperti Lynx.

HTTP pertama kali di luncurkan mengguanakan versi 1.0, sekarang disempurnakan menjadi very 1.1. Perbedaan HTTP/1.0 dengan HTTP/1.1 :

* HTTP/1.0 membuka satu koneksi untuk setiap permintaan URI

Header = Connection : close

* HTTP/1.1 dapat menggunakan sebuah koneksi TCP untuk beberapa permintaan URI(persistent)

Header = Connection: Keep-Alive

Kecuali jika client menyatakan tidak menggunakan hubungan persisten

(header=Connection: close)

HTTP status codes

* 200 OK
* 201 created
* 202 accepted
* 204 no content
* 301 moved perm
* 302 moved temp
* 304 not modified
* 400 bad request
* 401 unauthorized
* 403 forbidden
* 404 not found
* 500 int. server error
* 501 not impl.
* 502 bad gateway
* 503 svc not avail

# ALAT DAN LANGKAH PERCOBAAN

Alat yang digunakan adalah sebagai berikut:

1. Laptop
2. IDE Java Eclipse

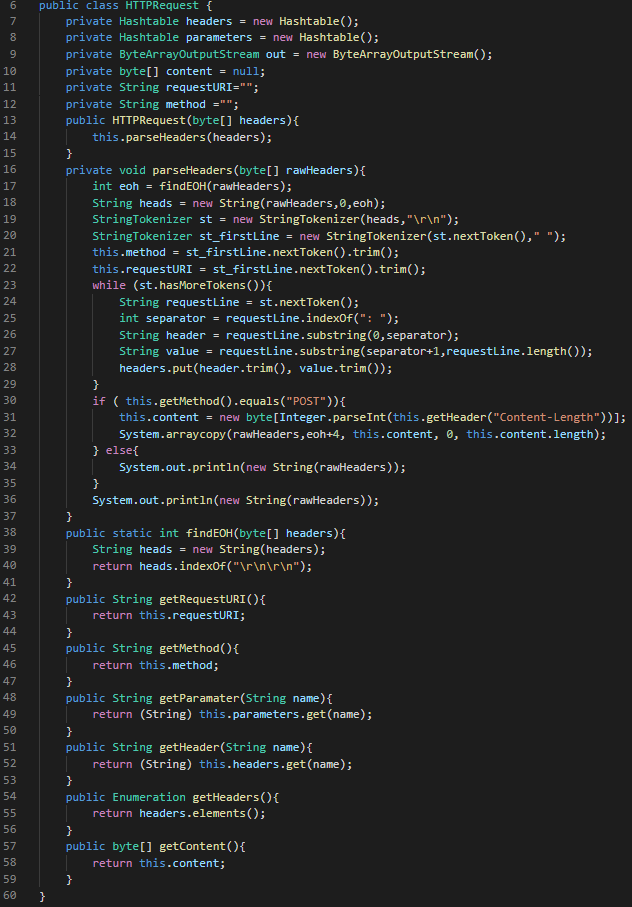
Langkah dalam melakukan pemrograman HTTP yaitu:

1. Membuat HTTPRequest.java; HTTPResponse.java; HTTPClient.java; dan HTTPServer.java
2. Membuat file index.php
3. *Running* file *.java*
4. Mengetikkan URL : [*http://localhost:8080/index.php*](http://localhost:8080/index.php)pada web browser

# Hasil dan Analisis Percobaan

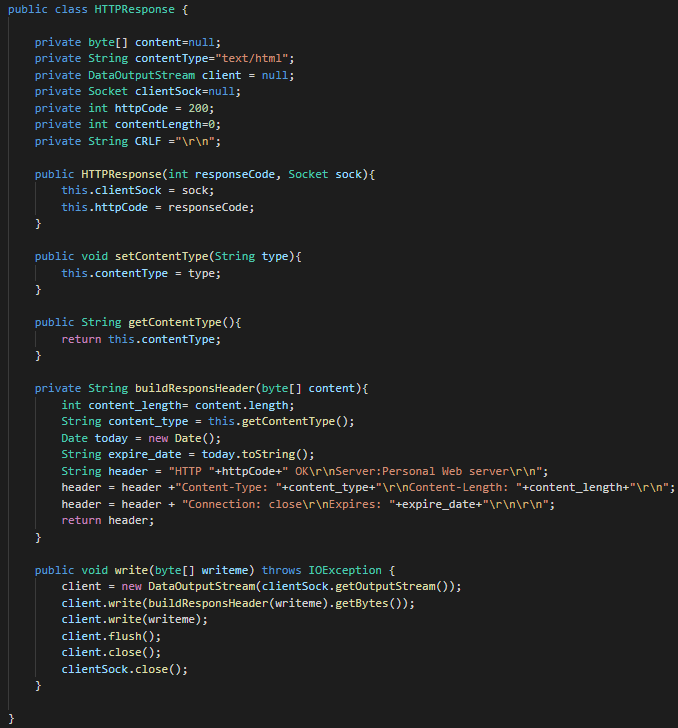
Program terdiri dari 4 file utama yaitu *HTTPRequest.java, HTTPResponse.java, HTTPServer.java*, dan *HTTPClient.java*.

1. HTTPRequest.java



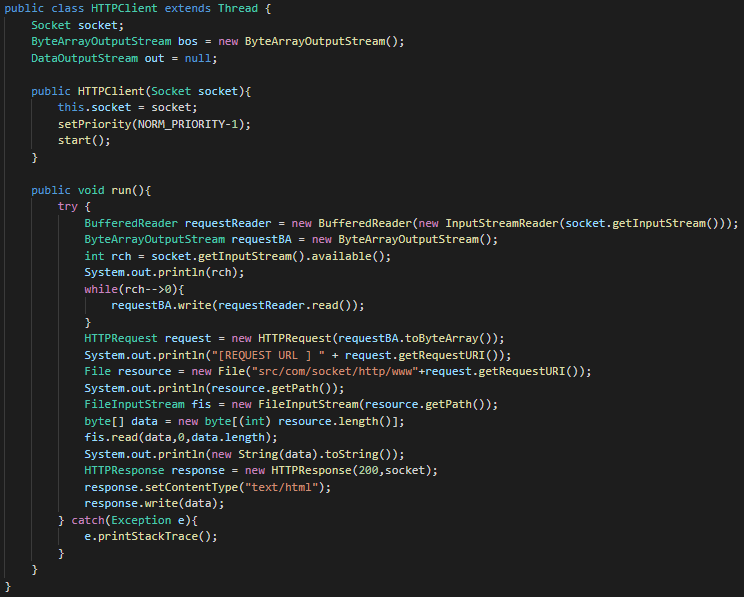
Di dalam code ini pertama dibuat objek baru dari Hashtable(); dan ByteArrayOutputStream(). Kemudian menginisiasi request URI awal sebagai null. Kemudian dibuat method untuk menerima atau membaca request dari client yang berupa verb yang diubah menjadi kode request yang diminta oleh client terhadap server. Di sini request dari client didapat dalam bentuk String yang kemudian diubah dalam bentuk token untuk menjadi kode dari tiap request oleh client. Di dalam code ini ditambahkan separator yaitu “:”. Ketika berhasil menemukan file yang sesuai dengan request dari client, maka akan ditampilkan melalui HTTPClient.java.

2. HTTPResponse.java



Di dalam code ini dibuat sebuah method untuk membaca lalu menampilkan output berupa text/html dari hasil request oleh client. Pertama kali dijalankan yaitu menginisiasi beberapa objek yaitu: *byte[] content=null; contentType="text/html"; DataOutputStream client = null; Socket clientSock=null; int httpCode = 200;* dan *contentLength=0*. Kemudian, melalui method HTTPResponse akan mencocokkan *clientSock* dan *httpCode*-nya. Kemudian menampilkan request/konten dari client berupa text/html sesuai token request ketika *httpCode*-nya bernilai 200 atau yang artinya berstatus OK. Ketika berhasil merespon, maka akan ditampilkan status mengenai respons seperti *Content-Type, Content-Length, Connection Time dan Expire Date*-nya.

3. HTTPClient.java

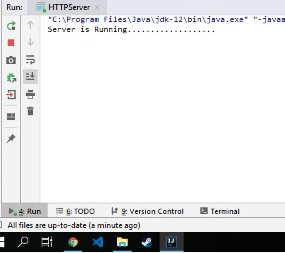


Dalam code ini, dibuat untuk *read* and *write* hasil dari request oleh client melalui browser. Pertama kali ketika file dijalankan, maka akan dibuat thread baru dengan halaman awal kosong. Method HTTPClient adalah untuk mencocokkan/mengambil socket dari client. Kemudian dalam method *run* dibuat objek baru dari BufferReader untuk membaca request URL dari client melalui HTTPRequest.java. Kemudian ketika berhasil mencocokkan kode respon melalui HTTPResponse.java maka akan ditampilkan file dari direktori *src/com/socket/http/www* dalam bentuk html.

4. HTTPServer.java

Di dalam code ini dibuat server menggunakan Thread dengan port 8080. Ketika server berhasil dijalankan akan muncul kalimat “Server is running...................”





# Kesimpulan

Dalam percobaan yang dilakukan, server berhasil menerima request dari client dan berhasil memberi response kepada client, yaitu melalui web browser mengakses URL yang kemudian mengeksekusi file *php* dalam bentuk html.

# DaftarPustaka

Winarno. 2019. *Modul Praktikum Sistem Terdistribusi”*.