

BAB II

TINJAUAN PUSTAKA

2.1 ARDUINO UNO R3

Mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus. Secara harfiahnya bisa disebut "pengendali kecil" dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi/diperkecil dan akhirnya terpusat serta dikendalikan oleh sebuah chip mikrokontroler.

Arduino adalah platform prototipe elektronik *opensource* yang berada dibawah lisensi *Creative Commons Attribution-ShareAlike 3.0*, berdasarkan perangkat keras dan lunak yang fleksibel dan mudah digunakan. Lisensi Creative Commons Attribution-ShareAlike 3.0 merupakan lisensi *opensource* yang memberikan jaminan kebebasan untuk mendistribusikan padapihak umum, mengubah dan membuat dokumentasi lengkap terhadap pengembangan piranti board arduino. Mikrokontroler nya diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan syntax dengan pemrograman C.

Revisi 3 dari board Arduino UNO memiliki fitur-fitur baru sebagai berikut:

- Pinout 1.0: ditambah pin *SDA dan SCL* yang dekat dengan pin *AREF* dan dua pin baru lainnya, yang diletakkan dekat dengan pin *RESET*, *IOREF* yang memungkinkan shield-shield untuk menyesuaikan tegangan yang disediakan dari board. Untuk kedepannya, shield akan dijadikan kompatibel/

cocok dengan board yang menggunakan AVR yang beroperasi dengan tegangan 3.3V (yang merupakan sebuah pin yang tak terhubung).

- Sirkuit RESET yang lebih kuat
- ATmega 16U2 menggantikan 8U2

2.1.1 Daya (Power)

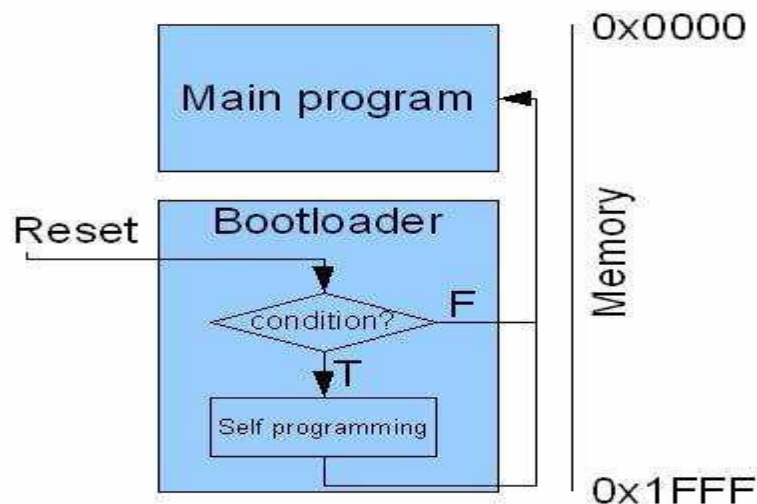
Arduino UNO dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau battery. Adaptor dapat dihubungkan dengan mencolokkan sebuah center-positive plug ke power jack dari board. Kabel lead dari sebuah battery dapat dimasukkan dalam header/kepala pin Ground (Gnd) dan pin Vin dari konektor POWER.

Board Arduino UNO dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt. Jika disuplai dengan yang lebih kecil dari 7 V, maka pin 5 Volt mungkin mensuplai kecil dari 5 Volt dan board Arduino UNO bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, voltage regulator bisa kelebihan panas dan membahayakan board Arduino UNO. Range yang direkomendasikan adalah 7 sampai 12 Volt.

2.1.2 Memori

Arduino dapat dikembangkan dari minimum system ATmega8/168/328 dengan mengisinya dengan bootloader. Bootloader memudahkan board arduino untuk berkomunikasi secara 2 level yakni

level atas adalah software Arduino IDE dan level bawah adalah sketch yang menjadi main program yang berinteraksi langsung dengan device. Bootloader dapat menerjemahkan sketch atau code program yang dibuat didalam software IDE yang nantinya akan diupload ke microcontroller itu sendiri.



Gambar 2.1 Arsitektur logical arduino

Bootloader hanya disimpan pada area tertentu di flash memory microcontroller. Misalnya pada mikro ATmega328 mengalokasikan 32KB untuk bootloader, 2 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis (RW/read and written) dengan EEPROM library).

2.1.3 Input dan Output

Setiap 14 pin digital pada Arduino Uno dapat digunakan sebagai input dan output, menggunakan fungsi pinMode(), digitalWrite(), dan digitalRead(). Fungsi-fungsi tersebut beroperasi di tegangan 5 Volt.

Setiap pin dapat memberikan atau menerima suatu arus maksimum 40 mA dan mempunyai sebuah resistor pull-up (terputus secara default) 20-50 kOhm.

Arduino UNO mempunyai 6 input analog, diberi label A0 sampai A5, setiapnya memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara default, 6 input analog tersebut mengukur dari ground sampai tegangan 5 Volt, dengan itu mungkin untuk mengganti batas atas dari rangenya dengan menggunakan pin AREF dan fungsi `analogReference()`.

fungsi-fungsi pin pada arduino :

Tabel 2.1 fungsi pin arduino

Nama Pin	Fungsi
VIN.	Tegangan input ke Arduino board ketika board sedang menggunakan sumber suplai eksternal (seperti 5 Volt dari koneksi USB atau sumber tenaga lainnya yang diatur). Kita dapat menyuplai tegangan melalui pin ini, atau jika penyuplaian tegangan melalui power jack, aksesnya melalui pin ini.
5V	Pin output ini merupakan tegangan 5 Volt yang diatur dari regulator pada board. Board dapat disuplai dengan salah satu suplai dari DC power jack (7-12V), USB connector (5V), atau pin VIN dari board (7-12). Penyuplaian tegangan melalui pin 5V atau 3,3V membypass regulator, dan dapat membahayakan board. Hal itu tidak dianjurkan.
3,3 V	Sebuah suplai 3,3 Volt dihasilkan oleh regulator pada board. Arus maksimum yang dapat dilalui adalah 50 mA
GND.	Pin Ground
Serial: 0 (RX) dan 1 (TX)	Digunakan untuk menerima (RX) dan memancarkan (TX) serial data TTL (Transistor-Transistor Logic). Kedua pin ini dihubungkan ke pin-pin yang sesuai dari chip Serial Atmega8U2 USB-ke-TTL.
External Interrupts: 2 & 3	Pin-pin ini dapat dikonfigurasi untuk dipicu sebuah interrupt (gangguan) pada sebuah nilai rendah, suatu kenaikan atau penurunan yang besar, atau suatu perubahan nilai.

PWM: 3, 5, 6, 9, 10, 11	Memberikan 8-bit PWM output dengan fungsi <code>analogWrite()</code> .
SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).	Pin-pin ini mensupport komunikasi SPI menggunakan SPI library.
LED: 13	Ada sebuah LED yang terpasang, terhubung ke pin digital 13. Ketika pin bernilai HIGH LED menyala, ketika pin bernilai LOW LED mati.
TWI: pin A4/SDA & pin A5/SCL	Mensupport komunikasi TWI dengan menggunakan Wire library.
AREF	Referensi tegangan untuk input analog. Digunakan dengan <code>analogReference()</code>
RESET	Membawa saluran ini LOW untuk mereset mikrokontroler. Secara khusus, digunakan untuk menambahkan melindungi yang memblock sesuatu pada board

2.1.4 Komunikasi

Arduino UNO mempunyai sejumlah fasilitas untuk komunikasi dengan sebuah komputer, Arduino lainnya atau mikrokontroler lainnya. ATmega 328 menyediakan serial komunikasi UART TTL (5V), yang tersedia pada pin digital 0 (RX), dan 1 (TX). Sebuah ATmega 16U2 pada channel board serial komunikasinya melalui USB dan muncul sebagai sebuah port virtual ke software pada komputer. Firmware 16U2 menggunakan driver USB COM standart, dan tidak ada driver eksternal yang dibutuhkan.

Bagaimanapun, pada windows sebuah file inf pasti dibutuhkan. Software Arduino mencakup sebuah serial monitor yang memungkinkan data tekstual terkirim ke dan dari board arduino. LED RX dan TX (pin 0 dan 1) pada board akan menyala ketika data sedang ditransmit melalui chip USB-to-serial dan koneksi USB pada komputer (tapi tidak untuk komunikasi serial). ATmega328 juga mensupport

komunikasi I2C (TWI) dan SPI. Software arduino mencakup sebuah Wire library untuk memudahkan menggunakan bus I2C.

2.1.5 Switch

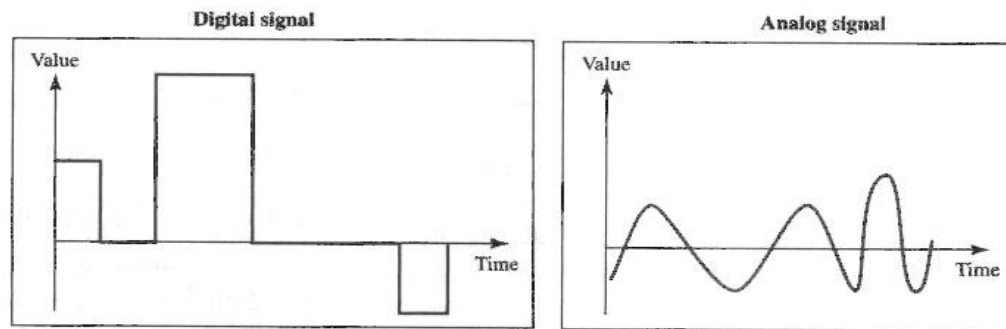
Switch difungsikan sebagai input digital dimana memiliki kondisi logika 1 dan 0. Kondisi ini mewakili perubahan level HIGH bernilai 1 dan LOW bernilai 0. Ketika sebuah switch berada dalam kondisi OFF, system tidak menerima input atau dengan kata lain berada dalam logika LOW. Jika switch ditekan pada kondisi ON, maka mendadak input memiliki nilai perubahan 1 dengan logika HIGH.



Gambar 2.2 simbol switch

2.1.6 Pulse Width modulation (PWM)

Pada dasarnya microcontroller berfungsi untuk mengolah input menjadi output. Input atau masukan sebagai informasi yang akan diolah pada microcontroller menggunakan sensor dan switch. Input dan output direpresentasikan kedalam 2 jenis yaitu analog dan digital. Analog memiliki kondisi yang mengacu informasi yang bersifat kontinyu, sedangkan digital mengacu pada nilai yang diskrit. Nilai diskrit yang dimaksud adalah memiliki kondisi yang pasti seperti contoh nilai 1 atau 0 dan nilai ON atau OFF.



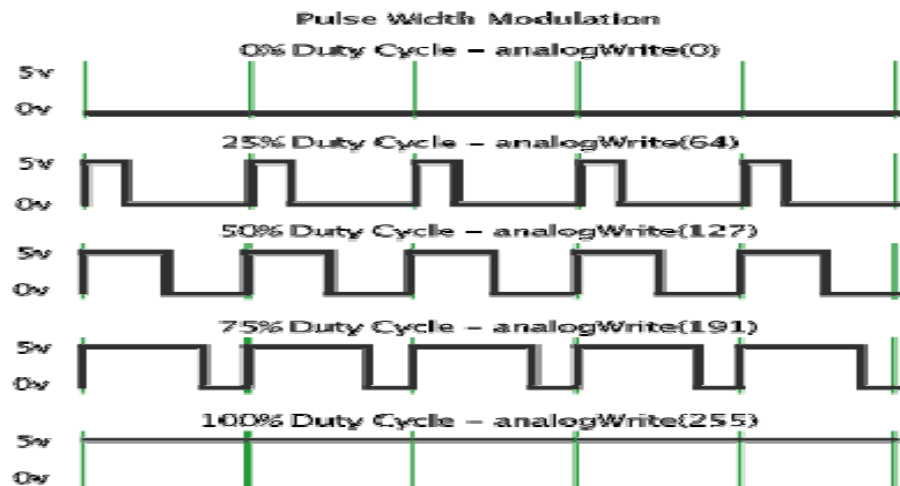
Gambar 2.3 Perbandingan sinyal digital dengan analog

Digital memiliki nilai yang terbatas karena bersifat diskrit, nilai tersebut dapat diwakilkan dengan perubahan level dalam jangka waktu tertentu. Nilai analog memiliki sinyal tak terbatas pada perubahan waktu tertentu berubah secara kontinu. Pada input sebuah sistem rangkaian elektronika, analog input dapat dihasilkan dari sensor sedangkan input digital dihasilkan dari switch.

PWM merupakan suatu metode mengontrol dan mendapat hasil analog dengan perangkat digital. Perangkat digital yang dimaksud adalah pembangkit gelombang persegi, sinyal berubah dengan pola ON-OFF. Pada Arduino, Pola on-off dapat mensimulasikan tegangan posisi penuh pada (5 Volt) dan off (0 Volt) dengan mengubah sebagian dari durasi waktu ketika posisi penuh 5V (on) dengan posisi off (0 V). Ketika durasi posisi 'on' disebut sebagai lebar pulsa, untuk mendapatkan variasi nilai analog dapat diubah masukan lebar pulsanya. Misalnya pola on-off dengan perubahan waktu yang sangat cepat dari tegangan 0 sampai 5v dapat mengubah brightness dari LED.

Dalam grafik di bawah, garis-garis hijau merupakan periode waktu yang teratur. Durasi ini merupakan periode yang memiliki nilai kebalikan dari frekuensi PWM. Dengan kata lain, dengan frekuensi PWM Arduino

di sekitar 500Hz, garis hijau akan mengukur periode waktu 2 milidetik. Fungsi untuk `analogWrite()` adalah pada skala 0-255 sehingga `analogWrite(255)` memiliki siklus 100% dan `analogWrite(127)` adalah siklus kerja 50% (pada separuh waktu).



Gambar 2.4 Konsep PWM dalam arduino

2.1.7 Kelebihan Arduino Sebagai Opensource Hardware

Dari beberapa hal yang berkaitan dari arduino sebagai opensource hardware untuk board edukasi memiliki beberapa kelebihan dibandingkan dengan board trainer elektronika yang dikembangkan :

1. Arduino menggunakan software yang berbasis IDE sehingga memudahkan pengguna awam untuk belajar bahasa pemrograman level microcontroller.
2. Pada level hardware, arduino memiliki bootloader sebagai program kecil yang ditanamkan dalam microcontroller ATmega dimana berfungsi untuk melakukan interaksi ke level atas dengan software IDE ketika program di compile dan di upload. Sedangkan pada level bawah,

bootloader dapat mengeksekusi sketch yang telah di upload pada mikrokontroller.

3. Multi platform, software arduino IDE dapat dijalankan pada semua jenis system operasi seperti Windows, Mac maupun Linux. Tidak hanya itu drivernya menggunakan chipset FTDI sebagai koneksi USB telah mendukung pada semua system operasi.
4. Tersedia kumpulan library dalam bentuk java, C/C++ siap pakai dan juga mampu digunakan dengan banyak perangkat input/output seperti sensor, LCD, 7-segmen maupun Ethernet.
5. Compatible dengan banyak shield sebagai modul pendukung fungsi pengembangan board arduino untuk proyek piranti elektronika.

2.2 Shield Arduino

Shield arduino dibuat untuk memperluas fungsi penggunaan untuk keperluan perangkat elektronika lain. Seperti contoh arduino tidak mampu menggerakkan motor stepper dan motor servo (driver motor), karena arduino tidak mampu mengolah arus listrik yang besar. Untuk itu dibutuhkan perangkat lain yang bisa digunakan sebagai penggerak motor dimana arduino berfungsi sebagai control.

Shield merupakan pengembangan dari perangkat elektronika umum yang digunakan dengan pengembangan library C/C++ dan bentuk dari referensi model board arduino, shield dapat digunakan dan compatible dengan board arduino. Berikut ini adalah beberapa contoh dari shield yang dikembangkan baik secara personal maupun pabrikasi untuk arduino :

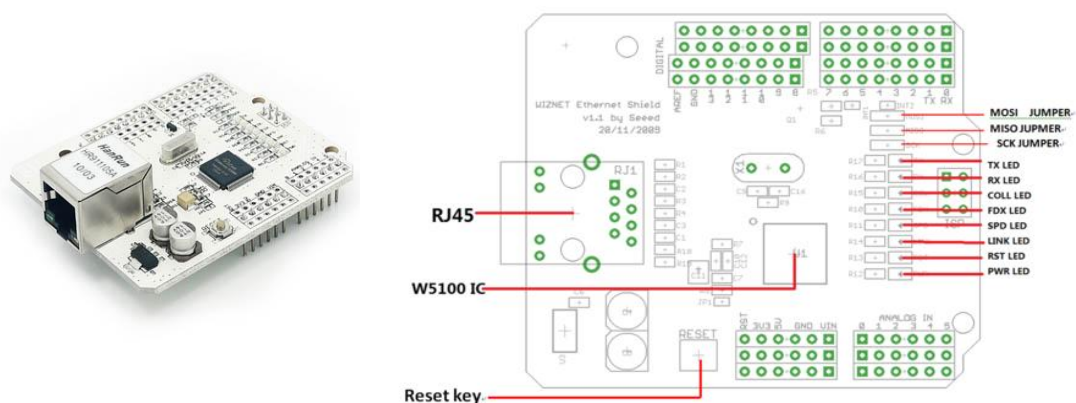
1. Arduino Motor Shield

Shield Motor untuk arduino mendukung arduino dengan motor driver menggunakan library Servo.h. Arduino menggunakan motor shield dapat digunakan untuk kebutuhan Robot bergerak dan tercontrol seperti Line follower robot, autonomus mobile robot, homemade CNC machine dan beberapa system mekatronika yang menggunakan motor.

2. Wirelesss Shield (Xbee/ZigBee)

Shield wireless arduino berupa modul Xbee digunakan sebagai pendukung proyek microcontroller berbasis wireless. Shield Xbee ini mendukung arduino digunakan untuk berkomunikasi secara wireless untuk membangun perangkat elektronika menggunakan protocol **802.15.4**

3. Ethernet Shield



Gambar 2.5 Ethenet Shield

Ethernet Shield berfungsi mengembangkan arduino berkomunikasi dengan internet. Shield yang dikembangkan menggunakan chipset Wiznet5100 yang telah mampu mendukung level TCP dan UDP. Penggunaan Shield ini dapat di

implementasikan dalam bentuk local area network maupun terhubung langsung dengan internet.

4. GPS/GSM shield

Shield GPS/GSM yang dikembangkan telah mampu mendukung arduino secara software(library) maupun hardware. Shield ini dapat digunakan dengan arduino untuk kebutuhan GPS tracker yang mampu di integrasikan dengan website google map, sedangkan GSM shield dapat dikembangkan sebagai SMS gateway berbasis arduino.

5. Proto Shield

Proto shield merupakan shield prototyping untuk arduino, menggunakan breadboard sebagai media penyusunan tata letak komponen sebagai kit percobaan. Protoboard bukanlah shield arduino dengan fungsi khusus seperti shield yang didesain final. Namun merupakan shield yang ditumpuk diatas arduino untuk memudahkan merancang percobaan rangkaian yang terintergrasi dengan arduino.

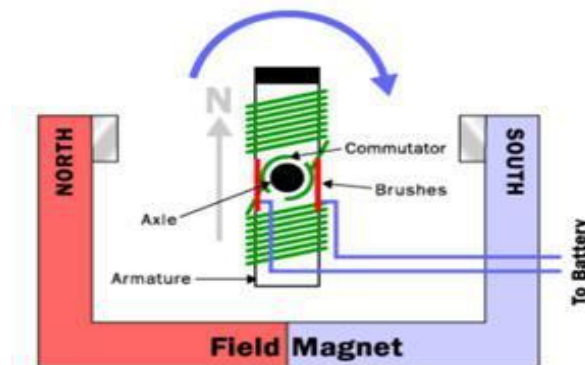
6. Compatible Shield lain

Compatible shield dikembangkan untuk mendukung lebih jauh kemampuan dan fungsionalitas arduino. Yang merupakan pengembangan shield arduino secara personal maupun massal untuk kebutuhan spesifik lain.

2.3 MOTOR DC

Motor DC adalah piranti elektronik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi. Pada motor DC terdapat jangkar dengan satu atau lebih kumparan terpisah. Tiap kumparan berujung pada cincin belah (komutator). Dengan adanya insulator antara komutator, cincin belah dapat berperan sebagai saklar kutub ganda (*double pole, double throw switch*).

Motor DC bekerja berdasarkan prinsip gaya Lorentz, yang menyatakan ketika sebuah konduktor beraliran arus diletakkan dalam medan magnet, maka sebuah gaya (yang dikenal dengan gaya Lorentz) akan tercipta secara ortogonal diantara arah medan magnet dan arah aliran arus. Mekanisme ini diperlihatkan pada gambar berikut ini.



Gambar 2.6 Bagan mekanisme kerja motor DC

Motor DC yang digunakan pada robot beroda umumnya adalah motor DC dengan magnet permanen. Motor DC jenis ini memiliki dua buah magnet permanen sehingga timbul medan magnet di antara kedua magnet tersebut. Di dalam medan magnet inilah jangkar/rotor berputar. Jangkar yang terletak di

tengah motor memiliki jumlah kutub yang ganjil dan pada setiap kutubnya terdapat lilitan.

Lilitan ini terhubung ke area kontak yang disebut komutator. Sikat (*brushes*) yang terhubung ke kutub positif dan negatif motor memberikan daya ke lilitan sedemikian rupa sehingga kutub yang satu akan ditolak oleh magnet permanen yang berada di dekatnya, sedangkan lilitan lain akan ditarik ke magnet permanen yang lain sehingga menyebabkan jangkar berputar. Ketika jangkar berputar, komutator mengubah lilitan yang mendapat pengaruh polaritas medan magnet sehingga jangkar akan terus berputar selama kutub positif dan negatif motor diberi daya. Kecepatan putar motor DC (N) dirumuskan dengan Persamaan berikut.

$$N = \frac{V_{TM} - I_A R_A}{K \phi}$$

Keterangan:

V_{TM} : Tegangan terminal

I_A : Arus jangkar motor

R_A : Hambatan jangkar motor

K : Konstanta motor

ϕ : Fluk magnet yang terbentuk pada motor.

Pengendalian kecepatan putar motor DC dapat dilakukan dengan mengatur besar tegangan terminal motor V_{TM} . Metode lain yang biasa digunakan untuk mengendalikan kecepatan motor DC adalah dengan teknik modulasi lebar pulsa atau **Pulse Width Modulation (PWM)**.

2.4 DASAR JARINGAN KOMPUTER

Router adalah sebuah alat jaringan komputer yang mengirimkan paket data melalui sebuah jaringan atau Internet menuju tujuannya, melalui sebuah proses yang dikenal sebagai routing. Proses routing terjadi pada lapisan 3 (Lapisan jaringan seperti Internet Protocol) dari stack protokol tujuh-lapis OSI. Router berfungsi sebagai penghubung antar dua atau lebih jaringan untuk meneruskan data dari satu jaringan ke jaringan lainnya.

2.4.1 Pengalamatan IP

Pengalamatan IP adalah pengindentifikasian dengan angka yang diberikan ke setiap mesin di dalam jaringan IP. Pengalamatan IP digunakan untuk menunjukkan lokasi spesifik dari alat di dalam jaringan. Alamat IP adalah alamat software, yang terpatris ke dalam Network Interface Card (NIC) dan digunakan untuk menemukan host pada jaringan lokal. Pengalamatan IP ditunjukkan untuk memungkinkan host didalam sebuah jaringan bisa berkomunikasi dengan Host pada jaringan yang berbeda, tanpa memperdulikan tipe dari LAN yang digunakan oleh host yang berpartisipasi.

Alamat IP terdiri atas 32 bit informasi. Bit ini terbagi menjadi 4 bagian, yang dikenal sebagai octet atau byte (8 bit). Pengalamatan IP dapat digambarkan dengan 3 metode:

- Dotted-decimal, seperti 172.16.30.56
- Biner, seperti 10101100.00010000.00010000.00011110.00111000
- Heksadesimal, seperti AC.10.1E.38

Tabel 2.2 Terminologi IP

Bit	Satu bit sama dengan satu digit; bernilai 1 atau 0
Byte	Satu Byte sama dengan 7 atau 8 bit, bergantung apakah menggunakan parity.
Octet	terdiri atas 8 bit, yang merupakan bilangan biner 8 bit umumnya.
Alamat network	digunakan dalam routing untuk menunjukan pengiriman paket ke remote network
Alamat broadcast	alamat yang digunakan oleh aplikasi dan host untuk mengirim informasi ke semua titik di dalam jaringan tersebut. Contoh 255.255.255.255 yang berarti semua jaringan, 172.16.255.255 yang berarti semua subnet dan host pada network 172.16.0.0 dan 10.255.255.255 yang berarti broadcast ke semua subnet dan host pada jaringan 10.0.0.0.

2.4.2 PENGALAMATAN NETWORK

Alamat network memberikan identifikasi unik untuk setiap jaringan. Setiap mesin pada jaringan yang sama menggunakan atau berbagi alamat network yang sama sebagai bagian dari pengalamatan IP. Pada alamat IP **172.16.30.56**, alamat networknya adalah **172.16**.

Alamat node memberikan identifikasi secara unik pada setiap mesin di dalam network. Bagian dari alamat ini haruslah unik karena alamat node mengidentifikasikan sebuah mesin tertentu. Nomor ini juga bisa disebut sebagai alamat host.. pada alamat IP **172.16.30.56**, angka **30.56** adalah alamat node.

Perancang internet memutuskan untuk membuat class dari jaringan berdasarkan ukuran jaringan. Untuk membagi sebuah alamat IP menjadi alamat network dan node ditentukan oleh class sebuah jaringan.

Tabel2.3 Range alamat network Class

Range alamat network Class A	jaringan class A dibuat Untuk jumlah jaringan kecil yang memproses node yang sangat banyak. Perancangan skema alamat IP mengatakan bahwa bit pertama dari setiap byte pada alamat jaringan Class A harus selalu off atau 0. Ini berarti alamat Class A adalah semua nilai antara 0 dan 127 .
Range alamat network Class B	jaringan class B dibuat Untuk jumlah jaringan yang berada diantara sangat besar dan sangat kecil seperti yang sudah diperkirakan. Pada jaringan Class B, RFC menyatakan bahwa bit pertama dari byte pertama harus selalu dalam kondisi on, tapi bit kedua harus dalam kondisi off. Ini berarti jaringan Class B ditentukan ketika byte pertama bernilai dari 128 sampai 191
Range alamat network Class C	jaringan class C dicanangkan Untuk jumlah jaringan yang banyak sekali dengan jumlah node sedikit. Untuk jaringan Class C, RFC menentukan 2 bit pertama dari octet pertama dari octet pertama harus selalu dalam kondisi on (hidup), tapi bit ke tiga harus dalam kondisi off. Jadi jaringan class C dimulai dari 192 sampai 223 .
Range alamat network Class D dan E	Alamat diantara 244 dan 255 dicadangkan untuk jaringan Class D dan E. Class D (224 sampai 239) digunakan sebagai alamat multicast dan Class E (240 sampai 255) hanya digunakan untuk penelitian

1. Pengalamatan class A

Di dalam jaringan Class A, byte pertama digunakan untuk menunjukan alamat network, dan tiga byte sisanya digunakan untuk alamat node.

Format class A yaitu:

network. node. node. node

Sebagai contoh, pada alamat IP **49.22.102.70**, angka **49** adalah alamat network, dan **22.102.70** alamat node. Setiap mesin pada jaringan ini akan mempunyai alamat network 49.

Jaringan class A menggunakan satu byte, dengan bit pertama dari byte tersebut dicadangkan dan 7 bit sisanya tersedia untuk dimanipulasi (pengalamatan). Sebagai hasilnya, jumlah maximum dari jaringan

Class A yang bisa buat adalah 128. Karena setiap posisi dari 7 bit bisa bernilai 0 atau 1, jadi jumlahnya adalah 2 pangkat 7 atau 128.

Alamat network yang terdiri atas 0 semua (0000 0000) dicadangkan untuk menunjukan rute default. Kemudian alamat 127 yang dicadangkan untuk keperluan diagnostik, juga tidak bisa digunakan, yang artinya hanya bisa menggunakan angka 1-126 untuk pengalamatan jaringan Class A adalah 128 dikurang 2, atau 126.

Setiap alamat Class A mempunyai 3 byte (tempat untuk 24 bit) untuk digunakan sebagai alamat node dari sebuah mesin. Ini artinya terdapat 2 pangkat 24 atau 16.777.216 kombinasi unik dan karenanya tepat sekali bahwa banyak sekali kemungkinan alamat node unik untuk setiap jaringan Class A. Karena alamat node dengan dua pola yaitu semua 0 dan semua 1 dicadangkan, jumlah maksimum node yang bisa digunakan untuk jaringan Class A adalah 2 pangkat 24 kurang 2, yang berarti setara dengan 16.777.214. Jumlah tersebut merupakan jumlah yang sangat besar untuk host di dalam sebuah segmen jaringan.

- *Host ID Class A yang Sah*

Host yang sah adalah host dengan angka diantara alamat network dan broadcast **10.0.0.1** sampai **10.255.255.254**. perhatikan bahwa angka 0 dan 255 bisa menjadi host ID yang sah. Yang perlu diingat ketika mencari alamat host yang sah adalah bit host tidak bisa off semua atau on semua pada waktu yang sama. Berikut ini contoh cara menentukan host ID yang sah di dalam pengalamatan jaringan Class A.

1. Semua bit host off, menunjukkan alamat network: **10.0.0.0**

2. Semua bit host on, menunjukkan alamat broadcast **10.255.255.255**

2. Pengalamatan Class B

Pada jaringan Class B, dua byte pertama menunjukkan alamat network dan dua byte. Selebihnya digunakan untuk alamat node. Formatnya yaitu:

network. network. node. node

Sebagai contoh, pada alamat IP **172.1630.56**, alamat network adalah **172.16** dan alamat node adalah **30.56**.

Dengan alamat network dua byte (masing-masing 8 bit), terdapat 2 pangkat 16 kombinasi unik. Namun perancang internet memutuskan bahwa semua jaringan Class B harus dimulai dengan digit 1 kemudian 0. Ini menyisakan 14 bit untuk dimanipulasi, karenanya terdapat 16.384 (yaitu 2 pangkat 14) alamat jaringan unik Class B.

Pengalamatan Class B menggunakan dua byte untuk pengalamatan node. Ini artinya 2 pangkat 16 di kurang dua pola yang dicadangkan (semua 0 dan semua 1) menjadi total 65.534 alamat node yang mungkin untuk setiap jaringan Class B.

- *Host ID Class B yang sah*

Host dengan angka di antara alamat network dan broadcast **172.16.0.1** sampai **172.16.255.254**. Berikut ini contoh cara menentukan host ID yang sah di dalam pengalamatan jaringan Class B.

1. Semua bit host off, menunjukkan alamat network: **172.16.0.0**

2. Semua bit host on, menunjukkan alamat broadcast: **172.16.255.255**.

3. Pengalamatan Class C

Tiga byte pertama dari pengalamatan jaringan Class C digunakan untuk alamat network, dengan hanya menyisakan satu byte kecil untuk alamat node. Formatnya yaitu

network. network. network. node

Dengan menggunakan contoh alamat IP **192.168.100.102**, alamat network adalah **192.168.100** dan alamat node adalah **102**.

Pada pengalamatan jaringan Class C, posisi tiga bit pertama selalu bernilai biner 110. Perhitungannya adalah 3 byte atau 24 bit dikurang 3 tempat yang dicadangkan, menyisakan 21 tempat oleh karena itu, terdapat 2 pangkat 21 atau 2.097.152 kemungkinan jaringan Class C.

Setiap jaringan unik Class C mempunyai satu byte untuk digunakan sebagai alamat node. Ini menimbulkan 28 atau 256 dikurang dua pola yang dicadangkan untuk semua 0 dan semua 1, menjadi total 24 alamat node untuk setiap jaringan Class C.

- *Host ID Class C yang sah*

Host yang sah yaitu host dengan angka di antara alamat network dan broadcast: **192.168.100.1** sampai **192.168.100.254**. Berikut ini contoh cara menentukan host ID yang sah di dalam pengalamatan jaringan Class C.

1. Semua bit host off, menunjukkan alamat network **192.168.100.0**

2. Semua bit host on, menunjukkan alamat broadcast **192.168.100.255**

Tabel 2.4 alamat IP yang dicadangkan

Class alamat	Alamat yang dicadangkan
Class A	10.0.0.0 sampai dengan 10.255.255.255
Class B	172.16.0.0 sampai dengan 172.31.255.255
Class C	192.168.0.0 sampai dengan 192.168.255.255

2.4.3 KONFIGURASI ROUTING IP

Router-router yang sudah terkonfigurasi hanya mempunyai informasi tentang network-network yang terhubung secara langsung. Informasi ini terdapat di masing-masing routing table. Dan jika sebuah router menerima sebuah paket untuk sebuah network yang tidak terdaftar di routing table, router tidak akan mengirimkan sebuah broadcast untuk mencari network remote (router hanya akan membuangnya).

Ada beberapa cara untuk mengkonfigurasi routing table untuk memasukkan semua network di internetwork kecil sehingga paket dapat diteruskan atau di forward ke network lain. Dan apa yang terbaik untuk satu network belum tentu terbaik untuk network lain. Jenis-jenis routing yang berbeda akan membantu mendapatkan yang terbaik untuk lingkungan tertentu dan sesuai kebutuhan masing-masing. Berikut ini adalah jenis-jenis routing :

1. Routing statis

Routing statis terjadi jika secara manual menambahkan route di routing table dari setiap router.

Routing statis memiliki keuntungan-keuntungan sebagai berikut:

- Tidak ada overhead (waktu pemrosesan) pada CPU router, yang berarti memungkinkan dapat membeli router yang lebih murah dari pada jika menggunakan routing dinamis.
- Tidak ada bandwidth yang digunakan diantara router, yang berarti memungkinkan dapat menghemat uang untuk link WAN.
- Routing statis menambah keamanan, karena administrator dapat memilih untuk mengizinkan akses routing ke network.

Routing statis memiliki kerugian-kerugian sebagai berikut:

- Administrator harus benar-benar memahami internetwork dan bagaimana setiap router dihubungkan untuk dapat mengkonfigurasi router dengan benar.
- Jika sebuah network ditambahkan ke internetwork, administrator harus menambahkan sebuah route ke semua router secara manual.
- Routing statis tidak sesuai untuk network-network yang besar.

2. Routing Default

Routing default digunakan untuk mengirimkan paket-paket ke sebuah network tujuan yang remote yang tidak ada di routing table, ke router hop berikutnya. Routing default hanya dapat digunakan pada network-network stub, yaitu network yang hanya memiliki satu jalur keluar (exit path) dari network itu. Untuk mengkonfigurasi sebuah route default, digunakan wild cards di alamat network dan lokasi mask dari

sebuah route statis. Bahkan sebenarnya, sebuah route default dapat dianggap sebuah route statis yang menggunakan wildcards.

3. Routing Dinamis

Routing dinamis adalah ketika routing protokol digunakan untuk menemukan network dan melakukan update routing table pada router. routing dinamis lebih mudah daripada menggunakan routing statis dan default, namun akan membebani dalam hal proses-proses di CPU router dan penggunaan bandwidth dari link jaringan. Sebuah routing protocol mendefinisikan kumpulan peraturan yang digunakan oleh router ketika router berkomunikasi tentang informasi routing dengan router-router yang lain.

2.5 PROTOKOL HTTP

Protocol HTTP berada pada layer application yang berfungsi sebagai sistem terdistribusi, kolaboratif dan hypermedia informasi. Protocol HTTP merupakan landasan utama untuk layanan Web/WWW, sehingga protocol HTTP merupakan inti dari komunikasi data pada layanan web.

HTTP di implementasikan kedalam bentuk fungsi *request* – *response* berupa model jaringan client – server. Client merupakan model yang biasanya melakukan request terhadap server, sedangkan server melakukan response sertamelayani kebutuhan client. Contoh dapat dijelaskan dalam terminology berikut :

Sebuah halaman web yang terdiri dari dokumen berupa obyek seperti fileHTML, gambar (baik JPEG maupun GIF), audio/video yang berada dalam alamatURL. Karena pada dokumen HTML tersebut biasanya disusun dari dari banyakobyek gambar, teks dan lainnya.

Contoh alamat : <http://www.someweb.com/index.html>

Alamat tersebut merepresentasikan isi website dengan halaman HTMLyaitu index.html yang disusun dari obyek teks dan gambar. Client melakukanrequest terhadap file dokumen index.html pada server melalui browser. Browseradalah agen dari client untuk menampilkan halaman web seperti Internet Explorerataupun Firefox. Kemudian server akan menerjemahkan permintaan clientterhadap isi dari index.html dan mengirimi satu persatu file dari index.html.Disisi server HTTP memiliki sebuah system yang disebut sebagai web serveryang melayani permintaan HTTP client. Web server seperti apache mampumenerjemah bahasa pemrograman level website seperti PHP, ASP , JSP dansebagainya.



Gambar 2.7 Mekanisme Request – Reponse Web HTTP

Model komunikasi HTTP terdiri dari web client sebagai browser yangmelakukan request halaman website dari web server. Model komunikasi client –server yang terjadi adalah HTTP request dari client sedangkan HTTP

responsedari server. Pada tahun 1997, browser dan web server diimplementasikan dengan HTTP/1.0. kemudian awal tahun 1998 semua browser dan web server di implementasikan dengan HTTP/1.1. Perkembangan protocol HTTP/1.1 masih mendukung versi protocol HTTP/1.0.

HTTP/1.0 dan HTTP/1.1 keduanya menggunakan layanan TCP pada transport layer. Untuk dapat melakukan request HTTP pada server, client terlebih dahulu melakukan koneksi TCP pada port 80 (HTTP) sebagai pintu masuk web server. Koneksi TCP akan diterima server dan kemudian mengijinkan pertukaran HTTP message berupa pertukaran data dari browser client dengan web server. Setelah pertukaran data terjadi, web server akan memutuskan koneksi. Jika pada waktu tertentu, client melakukan request pada obyek yang sama, web server akan melakukan proses reponse HTTP dengan mengirim kembali obyek tersebut.

Web server tidak mampu menyimpan kondisi akhir client saat request terjadi terhadap reponse. Kondisi ini dinamakan **stateless**, dimana HTTP server tidak mampu mengingat kondisi akhir request client.

2.5.1 Non-Persistent Connections

HTTP dapat digunakan kedalam bentuk persistent connections dan non persistent connections. Non-Persistent connections merupakan model yang digunakan untuk HTTP/1.0. Dan berikut ini merupakan mekanisme pengiriman halaman web dari server dalam kasus non-persistent connections.

Contoh : <http://www.someschool.edu/somedepartment/index.html>

Halaman web diatas diasumsikan dari **index.html** akan memanggilobyek gambar sebanyak 10 yang menyusun file HTML. Maka total obyek yang diminta oleh client adalah 11 yaitu file HTML berupa index dan 10 file gambar. Langkah – langkah mekanisme request obyek client terhadap server adalah sebagai berikut :

1. HTTP client akan menginisiasi koneksi TCP pada server<http://www.someschool.edu>. Port 80 pada server digunakan sebagai port default untuk mendengar permintaan client.
2. Server pada port 80 dan menjawab permintaan koneksi TCP client dengan mengirim pemberitahuan port 80 terbuka dan bisa diakses.
3. Client melakukan HTTP request terhadap halaman web/**somedepartment/index.html** dengan mengindikasikan HTTP message pada server
4. Server menjawab dan mengirim file index.html dan mengirim message pada koneksi TCP untuk menutup koneksi.
5. HTTP client menerima file index.html dan koneksi TCP terputus. Client menerjemahkan file HTML dan mencari referensi 10 obyek gambar yang menyusun HTML.
6. Client melakukan langkah 1 sampai 4 secara berulang hingga 10 obyek gambar yang menyusun file HTML lengkap. Dari mekanisme non-persistent, untuk meminta 11 obyek pada server maka client harus melakukan 11 koneksi TCP. Setiap koneksi TCP akan menginisiasikan request HTTP message pada sever dan setelah menjawab request HTTP, server akan menutup koneksi.

Pada kasus tertentu, untuk mempercepat permintaan obyek yang mengacu suatu HTML, client memiliki browser yang mampu melakukan proses parallel koneksi TCP. Dengan koneksi TCP yang bersifat parallel akan meminimalkan fungsi koneksi 1 TCP untuk 1 obyek sehingga client dapat meminta 10 obyek dalam waktu yang sama tanpa harus menunggu bergantian setiap 1 koneksi TCP.

Deskripsi mekanisme komunikasi yang terjadi antara client dan server dari request client sampai bagaimana paket diterima oleh client didefinisikan sebagai **round-trip time RTT**. Nilai RTT berkaitan dengan waktu pengiriman dari paket kecil yang dikirim oleh client sebagai request, paket yang dikirim oleh server sebagai reponse dan kembali pada client. RTT terdiri dari propagation delay, packet queuing delay pada router ataupun switch, dan packet processing delay. Ketika client melakukan klik hyperlink halaman web, hal ini akan memicu koneksi TCP antara browser client dengan web server.

2.5.2 Persistent Connections

Masalah yang terjadi pada non-persistent adalah membutuhkan 2RTT untuk request client yang terjadi. Untuk mempercepat request obyek yang mengacu file HTML, client melakukan proses parallel koneksi TCP. Masalah lain yang muncul, ketika server melakukan reponse maka harus mengalokasi resource hanya untuk menangani 1 HTTP request dari client. Hal ini akan mengakibatkan persoalan di sisi server untuk menangani request HTTP client jika obyek yang dikirim mencapai ratusan.

Pada Persistent connections, server akan tetap membiarkan koneksi TCP terbuka walaupun telah mengirim obyek ke client. Dan dalam jangka tertentu memiliki limit untuk memutuskan koneksi jika tidak ada lagi permintaan client yang bersifat simultan.

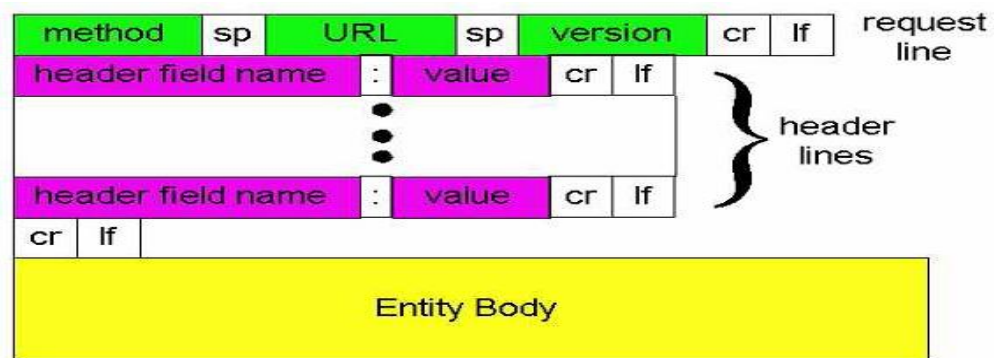
Selanjutnya untuk Request dan reponse yang sama dapat dikirim melalui 1 koneksi. Misalnya file HTML yang mengacu beberapa file dapat dikirim dalam 1 koneksi sampai semua file yang menyusun HTML tersebut lengkap, server kemudian dapat memutuskan koneksi. Ada 2 versi dalam persistent connections yaitu persistent dengan pipelining dan tanpa pipelining.

Versi persistent tanpa pipelining adalah client akan melakukan request obyek jika request obyek sebelumnya telah terpenuhi dan dijawab server dan setiap obyek hanya memerlukan 1 RTT. Permasalahan yang muncul adalah ketika server mengalami gangguan/hang tidak mampu menjawab request client maka client sama sekali tidak dapat untuk melakukan request kembali karena request sebelumnya belum terpenuhi.

Versi persistent dengan pipelining saat ini digunakan secara umum pada versi HTTP/1.1. Pada persistent dengan pipelining, client dapat melakukan request secara bersamaan pada obyek yang mengacu pada HTML.

2.5.3 HTTP Request Message

Berikut ini adalah format dari HTTP Request Message :



Gambar 2.8 Format Header request HTTP message

Baris pertama dinamakan **requestline**, biasanya mengidentifikasi sebagai method field, URL field, dan versi HTTP. Method field terdiri dari beberapa fungsi seperti yang termasuk didalamnya GET, POST, and HEAD.

Baris kedua dari format message mengidentifikasi sebagai **header lines**. **Connection: close**, hal ini bermaksud browser mengatakan pada server tidak melakukan koneksi persistent. Jika semua obyek telah dikirim, server boleh memutuskan koneksi. Browser mengimplementasikan HTTP/1.1 namun dalam message ini di generate tidak menggunakan koneksi persistent.

Pada baris selanjutnya (**User-agent: Mozilla/4.0**), mengidentifikasi browser client menggunakan Mozilla versi 4.0 sebagai user agent. Line **Accept: text/html, image/gif, image/jpeg**

dan **Acceptlanguage:fr** memberitahukan untuk menerima obyek text/html dan gambar berupa gif dan jpeg. Sedangkan Accept-language, header cenderung menerima obyek yang memiliki bahasa dalam versi French (fr).

Method dalam request message dibagi menjadi:

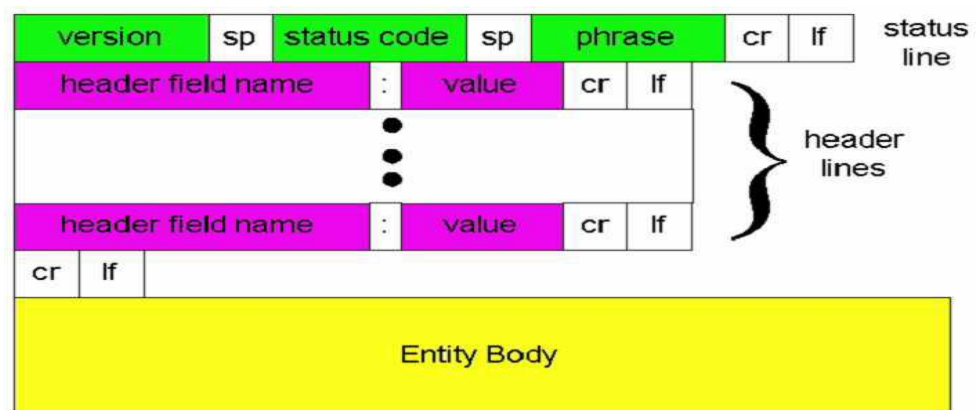
- HEAD ; fungsi method ini untuk menampilkan reponse dari server tanpa menampilkan body reponse. Biasanya digunakan untuk mengetahui meta-informasi dari suatu server tanpa mengirim content/obyek.
- GET ; request dengan method GET merupakan kondisi umum yang digunakan client untuk meminta informasi secara lengkap kepada server yang menyangkut halaman web.
- POST ; jika halaman web menampilkan form biasanya memerlukan input dari client. Method yang digunakan bertipe POST untuk mengidentifikasi sebagai request input dari client.
- PUT ; client dapat melakukan request untuk upload obyek pada resource server
- DELETE ; menghapus file pada server
- TRACE ; client dapat melakukan request cek perubahan file pada server
- OPTIONS ; fungsi ini sebagai tipe pengembalian untuk mengecek URL web server seperti dengan perintah wildcard '*' terhadap obyek tertentu
- CONNECT ; mengkonversi koneksi request dari mode biasa ke mode secure dengan fasilitas SSL (layanan HTTPS atau HTTP proxy)

- PATCH ; mengidentifikasi perbaikan terhadap resource

Tabel 2.5 Perbandingan Method Request pada HTTP/1.0 & HTTP/1.1

HTTP/1.0	HTTP/1.1
HEAD : menampilkan informasi server tanpa body	HEAD : menampilkan informasi server tanpa body
GET : mengambil obyek spesifik secara keseluruhan	GET : mengambil obyek spesifik secara keseluruhan
POST : mengirim data input dari client	POST : mengirim data input dari client
	PUT : mengupload file pada server
	DELETE : menghapus file pada server

2.5.4 HTTP Response Message



Gambar 2.9 Format header response HTTP message

Header line, merupakan identifikasi informasi server terhadap representasi internal data. Contoh **Connection: close** menyatakan pada client setelah pengiriman file maka koneksi akan diputus. **Date:** mendeskripsikan waktu response yang dilakukan oleh server dan server menampilkan versi web server yang digunakan. **Last modified** menampilkan kapan terakhir file tersebut diakses dan beberapa representasi data internal berkaitan dengan server. **Entire**

body merupakan data – data yang dikirim ke client sebagai payload.

Ada beberapa criteria dari code status response server terhadap client:

- **200 OK:** Request yang diminta ada, tipe pengembalian pengiriman obyek
- **301 Moved Permanently:** Request obyek yang diiminta telah dipindah, tipe pengembalian lokasi yang baru (Location:)
- **400 Bad Request:** Request tidak mengerti oleh server
- **404 Not Found:** Request terhadap obyek tidak ditemukan didalam server.
- **505 HTTP Version Not Supported :** Request yang diminta tidak sesuai dengan versi protocol HTTP server.