Monitoring Ruangan Menggunakan Webcam Server Pada Robot dan terintergrasi dengan laptop via wifi

Frangky Tupamahu ^{1*)},

Fajar Hermawanto ^{2*)}

Program Studi Teknik Informatika, Politeknik Gorontalo

email : frangky@poligon.ac.id,

email : fajar@poligon.ac.id

Ridwan Panigoro ³⁾

Program Studi Teknik Informatika, Politeknik Gorontalo

60111, email : ridwanpanigoro11@gmail.com

Sering kali pada saat melakukan monitoring yang memiliki tingkat kesulitan, seperti saat memonitoring sebuah ruangan dalam segi ukuran sulit dimasuki oleh manusia. Sehingga diperlukan suatu sistem, yang mampu melakukan monitoring pada sebuah ruangan atau keadaan sekitar robot. Robot ini berfungsi sebagai media pengganti dari manusia itu sendiri, dan memiliki kemampuan untuk berjalan atau beroperasi pada medan rata yakni dalam ruangan.

Monitoring Ruangan Menggunakan Webcam Server Pada Robot dan Terintegrasi Dengan Laptop Via Wifi adalah sebuah sistem monitoring menggunakan webcam server, yang terpasang pada sebuah robot yang mampu menampilkan gambar situasi sekelilingnya, yang ditangkap melalui webcam. Dari webcam, lalu gambar dikirim ke raspberry pi B+ dan memunculkan gambar kembali dilayar monitor yang telah terhubung dengan wireless melalui WiFi. Kemudian Signal wifi dipancarkan oleh acces point yang sudah terhubung dengan raspberry pi B+ dan hasilnya bisa dilihat secara jelas oleh pengguna.

Keywords : Monitoring, Webcam, Raspberry Pi B+, access point, robot.

I. PENDAHULUAN

Pada saat ini hampir tidak ada orang yang tidak mengenal robot, namun pengertian robot itu sendiri berbeda antara setiap orangnya karena kebanyakan orang membayangkan robot itu hanya sebuah mesin tiruan yang berbentuk manusia (*humanoid*). Namun bentuk robot bukan hanya dalam bentuk manusia (*humanoid*) saja karena masih banyak dalam bentuk lain misalnya seperti, hewan, kendaraan, dan lain-lain. Robot dapat didefinisikan sebagai sebuah alat mekanik yang dapat bekerja terus menerus untuk membantu pekerjaan manusia, yang dalam menjalankan tugasnya dapat dikontrol langsung oleh manusia maupun secara otomatis sesuai dengan program yang ditanamkan pada *chip controller* robot[5].

Sekarang ini kita sering melihat diberbagai media mengenai robot berteknologi canggih kemudian digunakan untuk mempermudah kegiatan manusia seperti pemantauan

bahkan melakukan pekerjaan sehari-hari. Biasanya ada pekerjaan yang dilakukan pada kondisi tertentu seperti pengambilan informasi visual pada sebuah ruangan yang sulit dilakukan oleh manusia secara langsung sehingga perlu adanya sistem yang mampu melakukan monitoring untuk

menampilkan informasi pada tempat yang ingin dimonitoring. Webcam juga dapat digunakan untuk sebagai sensor untuk pengambilan informasi serta menjadi indera penglihatan seperti mata yang ada pada manusia.

Berdasarkan kondisi diatas ada beberapa masalah yang timbul dalam pengambilan informasi karena memiliki tingkat kesulitan seperti memonitoring sebuah ruangan dalam segi ukuran sulit dimasuki oleh manusia Sehingga diperlukan suatu sistem yang mampu melakukan monitoring pada sebuah ruangan atau keadaan sekitar robot. Robot ini berfungsi sebagai media pengganti dari manusia itu sendiri dan memiliki kemampuan untuk berjalan atau beroperasi pada medan tertentu. Untuk mengendalikan robot menggunakan modul *remote control* dan untuk informasi visual menggunakan webcam yang terhubung ke *raspberry pi B+*.

II. METODE PENELITIAN

2.1 Perancangan Modul

2.1.1 Perancangan Modul Robot

Pada perancangan dan desain perangkat keras ini akan mulai dari rancangan mekanik robot dilakukan berdasarkan ukuran dan berat agar robot dapat bergerak dengan optimal. Robot yang akan digunakan merupakan robot beroda yang memiliki empat buah roda dengan bentuk menyerupai mobil.

2.1.2 Dimensi Robot

Robot yang akan dibuat memiliki ukuran sebagai berikut

:	
panjang	= 35 cm
lebar	= 25 cm
tinggi	= 17 cm

2.1.3 Material Robot

Adapun material untuk kerangka robot terbuat dari besi kotak dengan ukuran

4 cm x 2 cm, dan bagian roda mengadopsi roda sepeda anak-anak dengan jumlah 4 buah roda 2 di kiri dan di kanan sedangkan bagian-bagian mur dan baut, dibuat dari bahan baja. Untuk rancangan desain robot dapat dilihat pada gambar 3.1 dan 3.2 :

Gambar 3.1 Tampak Depan Robot

Gambar 3.2 Tampak Atas Robot

2.1.2 Perancangan Modul Remote Control

Remote control yang digunakan adalah *remote control* dari tank mainan. Modul *remote control* ini akan terhubung langsung dengan roda penggerak robot yakni motor DC. *Remote control* ini berfungsi sebagai *navigator* untuk mengontrol pergerakan robot.

2.1.3 Perancangan Modul Raspberry Pi B+

Perangan modul *raspberry* ini meliputi hubungan antara *Webcam* sebagai pengambil video, *Raspberry pi* sebagai otak dari sistem *Webcam*, *wireless router* sebagai penghubung antara laptop dengan kamera secara nirkabel.

2.2 Instalasi

2.2.1 Instalasi Sistem Operasi Raspberry Pi B+

Raspberry Pi memiliki sekitar 6 buah *distro* resmi sistem operasi masing-masing dengan keunikannya sendiri, yang pertama ada *NOOBS*, kemudian *RASPBIAN*, *PIDORA*, *OPNELEC*, *RASPBMC* dan *RISC OS*. Selain *distro* resmi, juga terdapat beberapa *distro* lain yang berbasis arsitektur ARM seperti *ArchLinux*. Untuk tugas akhir ini menggunakan sistem operasi *RASPIAN*.

2.2.2 Instalasi Aplikasi Motion

Aplikasi *motion* merupakan sebuah aplikasi sering digunakan untuk *Webcam server*. Aplikasi *motion* ini diinstall pada *raspberry* secara online dan masih memerlukan beberapa konfigurasi.

2.3 Konfigurasi

2.3.1 Konfigurasi Raspberry Pi B+

Konfigurasi yang harus dilakukan didalam *raspberry pi b+* seperti konfigurasi *IP Address* dan konfigurasi *Webcam*

2.3.2 Konfigurasi Acces Point

Konfigurasi pada *acces point* untuk melakukan pengaturan jaringan antara laptop dengan *raspberry Pi b+* yang terdiri dari *DHCP setting*, *wireless setting* dan *Address Reservation*.

2.3.3 Konfigurasi Aplikasi Motion

Pada aplikasi *motion* konfigurasi yang harus dilakukan yaitu menambahkan perintah pada *file motion*.

2.4 Gambaran Umum

Perancangan ini dilakukan untuk menggambarkan, merencanakan, dan membuat sketsa pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh pada robot dan pada *Raspberry pi B+*. untuk blok diagram sistem secara keseluruhan ditunjukkan pada gambar 3.3 :

Gambar 3.3 Blok Diagram Secara Keseluruhan

Penjelasan cara kerja dari blok diagram secara keseluruhan adalah sebagai berikut :

1. *Webcam* berfungsi untuk mengambil video yang akan di tampilkan pada laptop.
2. *Raspberry* berfungsi untuk tempat menghubungkan *Webcam* dan *acces point*.
3. *Acces Point* berfungsi untuk menghubungkan leptop dengan *raspberry* sehingga dapat mengakses *Webcam*.
4. *USB Wireless* berfungsi untuk menghubungkan *raspberry* secara nirkabel.
5. Laptop berfungsi untuk menampilkan hasil yang didapat oleh *Webcam* berupa video stream.
6. *Remote Control (Transmitter)* berfungsi sebagai pengirim perintah yang akan mengontrol gerak robot.
7. Modul *Remote Control (Receiver)* berfungsi sebagai penerima perintah dari *transmitter* dan langsung menggerakkan motor DC.
8. Motor DC berfungsi sebagai penggerak pada robot.

V. IMPLEMENTASI

3.1 Implementasi

Tujuan pembuatan sistem ini adalah mengimplementasikan desain sebuah sistem *webcam server* pada robot beroda yang dilengkapi *webcam* yang terintegrasi dengan laptop dan mampu menggantikan manusia sehingga mempermudah manusia untuk melakukan monitoring pada sebuah tempat yang sulit dijangkau oleh manusia.

3.2 Instalasi Raspian

Pada proses instalasi *raspian* ini sebelum melakukan instalasi perlu mengunduh sistem operasi *raspian* terlebih dahulu dan memasukan ke dalam *microSD* card.

3.3 Konfigurasi Ip address

Konfigurasi *Ip address* ini dilakukan untuk memberikan dan mengaktifkan alamat IP pada komputer sehingga dapat digunakan pada jaringan komputer. Untuk melakukan konfigurasi *ip address* pada raspberry menggunakan perintah seperti pada gambar 4.9 :

```
root@raspberrypi:/home/pi# nano /etc/network/interfaces
```

Gambar 4.9 Perintah Konfigurasi Ip Address

```
GNU nano 2.2.6 File: /etc/network/interfaces
auto eth0
iface eth0 inet static
    address 192.168.10.1
    netmask 255.255.255.252
    network 192.168.10.0
    broadcast 192.168.10.3
```

Gambar 4.10 Konfigurasi Ip Address

Setelah itu restart network dengan menggunakan perintah seperti yang ditunjukkan pada gambar 4.11 :

```
pi@raspberrypi ~ $ sudo /etc/init.d/networking restart
```

Gambar 4.11 Perintah Restart Network

Untuk melihat hasil konfigurasi yang sudah dibuat maka dengan perintah seperti pada gambar 4.12 :

```
pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:ab:d9:f7
          inet addr:192.168.10.1  Bcast:192.168.10.3  Mask:255.255.255.252
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:322 errors:0 dropped:0 overruns:0 frame:0
          TX packets:213 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:23909 (23.3 KiB)  TX bytes:24199 (23.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:216 errors:0 dropped:0 overruns:0 frame:0
          TX packets:216 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
```

Gambar 4.12 Perintah dan Hasil Ifconfig

3.4 Konfigurasi Motion

Pada tahap konfigurasi motion ini dimulai dengan menginstal sebuah paket, yaitu paket *motion*. *Motion* sendiri merupakan media yang digunakan untuk *monitoring* menggunakan *webcam* yang ada. Berikut ini adalah perintah *update* paket

agar sistem memiliki aplikasi yang selalu terbaru. Perintah *update* dapat dilihat pada gambar 4.13 :

```
pi@raspberrypi ~ $ sudo apt-get update
```

Gambar 4.13 Perintah Update Program

Setelah proses *update* selesai, pada tahap selanjutnya yaitu install paket *motion*. Perintah untuk menginstal paket *motion* seperti yang ditunjukkan pada gambar 4.14 :

```
pi@raspberrypi ~ $ sudo apt-get install motion
```

Gambar 4.14 Perintah Install Motion

Setelah proses instalasi selesai, proses selanjutnya adalah mengkonfigurasi *file motion.conf* perintah yang digunakan seperti yang ditunjukkan pada gambar 4.15:

```
pi@raspberrypi ~ $ sudo nano /etc/motion/motion.conf
```

Gambar 4.15 Perintah Untuk Mengedit File motion.conf

Untuk merubah isi *file* dalam *motion.conf* tekan *Crtl + W* untuk mencari baris yang akan diganti. Kemudian cari baris *daemon* setelah ditemukan ganti *status default daemon* dari *off* ke *on* hal ini dilakukan untuk membuat *motion* berjalan. Seperti yang ditunjukkan pada gambar 4.16 :

```
# Start in daemon (background) mode and release terminal (default: off)
daemon on
```

Gambar 4.16 Daemon On

Kemudian mencari baris *webcam localhost* dan pada status default *on* ganti dengan *off* hal ini dilakukan untuk memungkinkan dapat mengaksesnya koneksi dari luar. Langkahnya seperti yang ditunjukkan pada gambar no 4.17 :

```
# Restrict webcam connections to localhost only (default: on)
webcam localhost off
```

Gambar 4.17 Webcam Localhost

Kemudian mencari baris *control localhost* dan merubah status default dari *on* menjadi *off* hal ini dilakukan agar bisa mengontrol alat jaringan dari luar. Langkahnya seperti ditunjukkan pada gambar no 4.18 :

```
# Restrict control connections to localhost only (default: c
control_localhost off
```

Gambar 4.18 *Control_Localhost*

Selanjutnya untuk mengubah ukuran frame dari gambar yang akan ditampilkan yang akan dilakukan terlebih dahulu yaitu mencari baris *image width* untuk mengatur lebar dan *image height* untuk tinggi. Ukuran yang digunakan yaitu lebar 320 px dan tinggi 240 px. Seperti yang ditunjukkan pada gambar 4.19 :

```
# Image width (pixels). Valid range: Camera dependent, default: 352
width 320

# Image height (pixels). Valid range: Camera dependent, default: 288
height 240
```

Gambar 4.19 *Ukuran Frame (width and height)*

Setelah mengubah ukuran frame, proses selanjutnya yang akan dilakukan adalah mengubah *framerate*, *framerate* berfungsi untuk mengatur Jumlah maksimum frame yang akan diambil oleh kamera per detik. Pada *framerate* bisa diisi dengan nilai 2 sampai 100 untuk jumlah default yaitu 100. *Framerate* yang digunakan saat ini dengan nilai 30 frame per detik. Seperti yang ditunjukkan pada gambar 4.20 :

```
# Maximum number of frames to be captured per second.
# Valid range: 2-100. Default: 100 (almost no limit).
framerate 30
```

Gambar 4.20 *Framerate*

Selanjutnya mencari baris *quality* untuk menentukan kualitas yang dihasilkan oleh kamera dalam hitungan % . untuk setelan default 75 %, namun untuk alat monitoring ini menggunakan 100% . seperti yang ditunjukkan pada gambar 4.21 :

```
# The quality (in percent) to be used by the jpeg compression (default: 75)
quality 100
```

Gambar 4.21 *Quality*

Konfigurasi yang terakhir yaitu untuk membuat paket *motion* selalu aktif ketika *raspberry* baru dihidupkan, lakukan konfigurasi pada file */etc/default/motion*, ubah *start_motion_daemon=no* menjadi *start_motion_daemon=yes*. Seperti gambar 4.22 :

```
# set to 'yes' to enable the motion daemon
start_motion_daemon=yes
```

4.22 Ubah Default Motion

3.5 Konfigurasi DNS

Fungsi utama dari *DNS* adalah Menerjemahkan nama-nama host (*hostnames*) menjadi nomor IP (*IP address*) ataupun sebaliknya, sehingga nama tersebut mudah diingat oleh pengguna jaringan memberikan suatu informasi tentang suatu host pada jaringan lokal. Pemanfaatan *DNS* itu sendiri pada tugas akhir ini yaitu guna mempermudah pada saat pengaksesan *webcam server* ini secara lokal tanpa perlu mengetikkan *ip address* lagi. Sebelum melakukan konfigurasi, hal yang pertama dilakukan yaitu menginstall paket yang akan digunakan untuk menerjemahkan *ip address* menjadi sebuah domain. Paket yang diinstall yaitu *bind9*, cara install dan perintah konfigurasi *bind9* diraspberry seperti yang ditunjukkan pada gambar 4.23 :

```
pi@raspberrypi ~ $ sudo apt-get install bind9
```

Gambar 4.23 *Perintah Install Bind9*

Selanjutnya menyalin file *db.local* yang ada di dalam direktori */etc/bind/* dengan nama baru yaitu *db.web*. perintah yang digunakan untuk menyalin file *db.local* menjadi *db.web* seperti yang ditunjukkan pada gambar 4.24 :

```
pi@raspberrypi ~ $ sudo cp /etc/bind/db.local /etc/bind/db.web
```

Gambar 4.24 *Perintah Menyalin file db.local ke db.web*

Setelah selesai menyalin file, kemudian mengedit isi dari file *db.web* dengan menggunakan editor *nano*. Perintah untuk mengedit file menggunakan editor *nano* seperti yang ditunjukkan pada gambar 4.25 :

```
pi@raspberrypi ~ $ sudo nano /etc/bind/db.web
```

Gambar 4.25 *Perintah Edit file db.web*

Setelah itu yang perlu dilakukan adalah merubah isi file *db.web* yang telah disalin dengan mengubah beberapa *script*, seperti yang ditunjukkan pada gambar 4.26 :

```
GNU nano 2.2.6 File: /etc/bind/db.web

; BIND data file for local loopback interface
;
$TTL 604800
@ IN SOA webcam.server. root.webcam.server. (
    2 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS webcam.server.
@ IN A 192.168.1.2
@ IN AAAA ::1
ns IN A 192.168.1.2
www IN CNAME ns
```

Gambar 4.26 *Tampilan Script db.web*

Untuk menyimpan menggunakan perintah *Ctrl+O* kemudian tekan enter dan keluar dengan perintah *Ctrl+x*, kemudian setelah selesai langkah selanjutnya adalah *copy file* *db.127* menjadi *db.192*. Perintah untuk menyalin file *db.127* ke *db.192* seperti yang tunjukan pada gambar 4.27 :

```
pi@raspberrypi ~ $ sudo cp /etc/bind/db.127 /etc/bind/db.192
```

Gambar 4.27 Perintah Menyalin file db.127 ke db.192

Setelah menyalin file, selanjutnya merubah isi dari file db.192 menggunakan perintah seperti yang ditunjukkan pada gambar 4.28 :

```
GNU nano 2.2.6 File: /etc/bind/db.192
; BIND reverse data file for local loopback interface
;
$TTL 604800
@ IN SOA webcam.server. root.webcam.server. (
    1 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS webcam.server.
2 IN PTR webcam.server.
```

Gambar 4.28 Tampilan Script db.192

Untuk menyimpan digunakan perintah Ctrl+O kemudian tekan enter dan keluar dengan perintah Ctrl+x, langkah selanjutnya adalah mengedit file *named.conf.default-zones* Karena kita akan membangun dns server lokal, maka kita membuat *forward zone* dan *reverse zone* didalam file *named.conf.local*. Perintah yang digunakan untuk mengubah file *named.conf.default-zones* seperti yang ditunjukkan pada gambar 4.29 :

```
pi@raspberrypi ~ $ sudo nano /etc/bind/named.conf.default-zones
```

Gambar 4.29 perintah mengubah file *named.conf.default-zones*

Selanjutnya menambahkan isi file yang ditambahkan kedalam file *named.conf*. dengan menambahkan script seperti yang ditunjukkan pada gambar 4.30 :

```
zone "webcam.server" {
    type master;
    file "/etc/bind/db.web";
};

zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.192";
};
```

Gambar 4.30 Tampilan Script yang Ditambahkan pada file *named.conf.default-zones*

Untuk menyimpan digunakan perintah Ctrl+O kemudian tekan enter dan keluar dengan perintah Ctrl+x, selanjutnya edit file */etc/resolv.conf* dengan menggunakan perintah seperti yang ditunjukkan pada gambar 4.31 :

```
pi@raspberrypi ~ $ sudo nano /etc/resolv.conf
```

Gambar 4.31 Perintah edit file */etc/resolv.conf*

Setelah mengetikan perintah seperti pada gambar 4.31, kemudian memasukan beberapa script seperti yang ditunjukkan pada gambar 4.32 :

```
GNU nano 2.2.6 File: /etc/resolv.conf
# Generated by resolvconf
nameserver 192.168.1.2
search webcam.server
```

Gambar 4.32 Script *Resolv.conf*

Untuk menyimpan menggunakan perintah Ctrl+O kemudian tekan enter dan keluar dengan perintah Ctrl+x, selanjutnya *restart* paket *bind9* dan *networking* dengan mengetikan perintah seperti yang ditunjukkan pada gambar 4.33 :

```
pi@raspberrypi ~ $ sudo /etc/init.d/bind9 restart
```

Gambar 4.33 perintah untuk restart bind9

3.6 Konfigurasi Acces Point

Guna mengakses via *wireless*, digunakan sebuah *access point*. *Access point* sendiri merupakan perantara *user-user* untuk mengakses *webcamserver* dengan menjadikan *ip address access point* sebagai *gateway*. *Gateway* sendiri berfungsi untuk membubungkan satu jaringan komputer dengan satu atau lebih jaringan komputer yang menggunakan kaidah komunikasi yang berbeda sehingga informasi dari satu jaringan komputer dapat dialihantarkan ke jaringan komputer yang lain dengan kaidah jaringan berbeda. Konfigurasi *access point* sendiri dimulai dengan mengetikan *ip address access point* pada aplikasi browser seperti yang ditunjukkan pada gambar 4.40 :

Gambar 4.40 Pengisian Ip address Acces Point

Setelah mengisi *ip address* pada url browser dan menekan tombol enter kemudian akan muncul autentikasi untuk memasukan nama pengguna dan sandi, nama pengguna dan sandi yang digunakan adalah nama pengguna dan sandi default yaitu nama pengguna *admin* dan sandi *admin*, seperti yang tunjukkan pada gambar 4.41 :

Gambar 4.41 Autentikasi Acces Point

Setelah memasukan nama pengguna dan sandi maka akan masuk ke dalam Halaman *Web Access point*. Seperti yang ditunjukkan pada gambar 4.42 :

Gambar 4.42 Halaman Web Access point

Setelah masuk pada halaman web ada beberapa hal yang harus dikonfigurasi dimulai dari *quick setup*, setelah itu menekan tombol *next* seperti yang ditunjukkan pada gambar 4.43 :

Gambar 4.43 Tampilan Quick Setup

Setelah menekan tombol *next* maka masuk ke proses selanjutnya yaitu choose access point mode, ada 3 pilihan yang disediakan namun yang digunakan adalah mode AP (*Access Point*) seperti yang ditunjukkan pada gambar 4.44 :

Gambar 4.44 Tampilan Choose Operation Mode

Setelah memilih AP pada *access point mode* dan menekan *next*. Selanjutnya masuk pada *quick setup wireless*, pada *wireless* terdapat beberapa kolom yg terisi secara *default*. Kemudian diubah isi pada kolom seperti yang ditunjukkan pada gambar 4.45 :

Gambar 4.45 Quick Setup Wireless

Kemudian untuk melanjutkan ke proses selanjutnya dengan menekan *next*, maka akan masuk ke proses *quick setup finish* seperti yang ditunjukkan pada gambar 4.46 :

Gambar 4.46 Quick Setup Finish

Setelah menekan tombol *finish*, selanjutnya masuk pada konfigurasi DHCP. Konfigurasi DHCP ini bertujuan untuk mempermudah client untuk mengakses webcam server tanpa harus mengatur *ip address* lagi. Ada beberapa hal yang dikonfigurasi yaitu sebagai berikut :

1. *DHCP server* pilih *enable* untuk mengaktifkan *DHCP server*.
2. *Start ip address end ip address* untuk *range ip* dari 192.168.1.1 sampai 192.168.1.10
3. *Default gateway* 192.168.1.254 merupakan *ip address* dari *access point*.
4. *Primary DNS* 192.168.1.2 merupakan *ip address* dari *raspberry* atau *ip webcam server*.

Selanjutnya menekan *save* untuk menyimpan hasil konfigurasi, *DHCP setting* seperti yang ditunjukkan pada gambar 4.47 :

Gambar 4.47 Tampilan DHCP setting

Setelah menekan tombol maka sistem akan meminta restart seperti yang ditunjukkan pada gambar 4.48:

Gambar 4.48 DHCP Setting Meminta Untuk Reboot

3.7 Hasil dan Pengujian

Pengujian medan robot ini dilakukan pada sebuah *miniature* sebuah ruangan dengan lebar 45 cm yang sulit dimasuki oleh manusia. Robot dapat berjalan dengan baik pada *miniature* ruangan ini. Untuk mengujikan informasi apa yang didapatkan robot sekaligus menguji *webcam server* saat berada dalam ruangan digunakan sebuah bola dan gelas berwarna. Seperti yang ditunjukkan pada gambar 4.53 dan 4.54 :

DAFTAR PUSTAKA

- [1] Ali, M., 2009. Jurnal Universitas Negeri Yogyakarta : Pengembangan Media Pembelajaran Interaktif. Yogyakarta.
- [2] Fathurrohman M., 2012. Jurnal. Penggunaan Multimedia dalam Pembelajaran. Halaman 6
- [3] Lestari, D., 2012. Jurnal Arsip Teknik Informatika UMMI. Definisi Multimedia.
- [4] Mulyana, E., 2012. App Inventor : Ciptakan Sendiri Aplikasi Androidmu. Penerbit Andi. Halaman 2
- [5] Santosa, B., 2011. Repository UPI(Universitas Pendidikan Indonesia). Media Pembelajaran. Halaman 11-12
- [6] Sofyan, A. F. dan Purwanto, A., 2008 Digital Multimedia : Animasi, Sound Editing, & Video Editing. Yogyakarta : Andi Offset. Halaman 3-4
- [7] Supriatna, D., 2009. Jurnal : Pengenalan Media Pembelajaran. Hal 3 dan 4
- [8] Winarno, E. dan Zaki A., 2011. Hacking dan Programming dengan Android SDK untuk Advanced. Elex Media Komputindo. Halaman 5.