

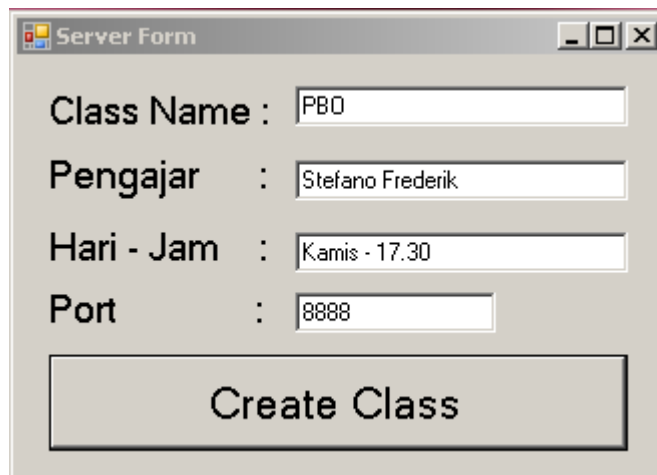
5. PENGUJIAN PROGRAM

Pada bab ini dibahas penggunaan aplikasi beserta pengujiannya. Penggunaan aplikasi akan dibagi menjadi dua, yaitu untuk *server* dan untuk *client*. Pengujian dilakukan pada *Local Area Network* (LAN). Hal-hal yang diujikan adalah segala sesuatu yang berhubungan dengan jaringan antara lain penggunaan *bandwidth* pada jumlah client tertentu pada suatu *class* ataupun jumlah *server* yang sedang *multicast*. Selain itu, *input* dari *server* yang berubah yang dalam hal ini adalah resolusi monitor yang dipakai.

5.1. Penggunaan Program

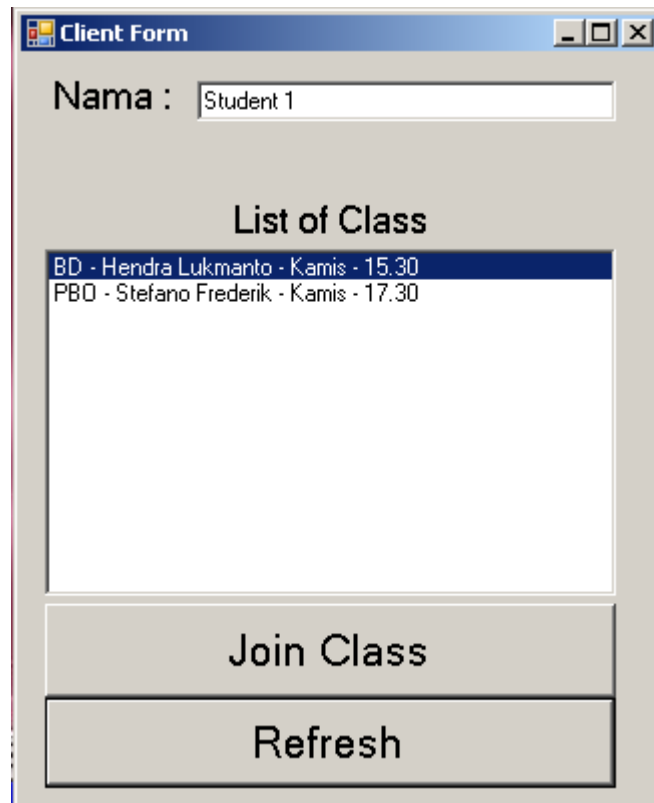
Penggunaan program akan dibagi menjadi dua bagian, yaitu bagian *server* dan *client*. *Server* adalah seorang pengajar dan *client* adalah seorang pelajar.

Dalam menggunakan aplikasi *server*, *user* pertama kali diharuskan mengisi beberapa *field* yaitu *Class Name*, Pengajar, Hari – Jam, dan *port*. Setelah mengisi *field* tersebut, maka *user* diperbolehkan untuk menekan tombol *Create Class* yang berada di bawah tengah dari *form*.



Gambar 5.1. Server Form untuk Create Class

Setelah menekan tombol *Create Class*, maka data yang telah dimasukkan akan dikirimkan kepada semua *client* yang sedang aktif. Tampilan seorang *client* dapat dilihat pada Gambar 5.2.



Gambar 5.2. Tampilan pada *Client* saat ada *Class*

Pada *client* ditampilkan semua *class* yang dapat diikuti. Ada kemungkinan ketika *server* membuat *class*, *client* belum aktif dan *client* tersebut tidak akan mendapat data tentang *class* yang telah dibuat sehingga pada *listbox* List of Class kosong. Untuk menanggulangi hal tersebut, *client* dapat menekan tombol Refresh untuk meminta *request* data semua *class* yang telah ada dan mendapat data tentang *class* yang telah dibuat seperti pada Gambar 5.2.

Sebelum mengikuti suatu *class*, *client* diharuskan untuk mengisi beberapa *field* yaitu Nama dan Port. Setelah itu *client* memilih akan mengikuti *class* yang mana dengan menekan tombol Join Class. Tampilan *client* apabila telah masuk ke dalam *class* dapat dilihat pada Gambar 5.3.

Client Form

Nama : Student 1

List of Class

BD - Hendra Lukmanto - Kamis - 15.30
PBO - Stefano Frederik - Kamis - 17.30

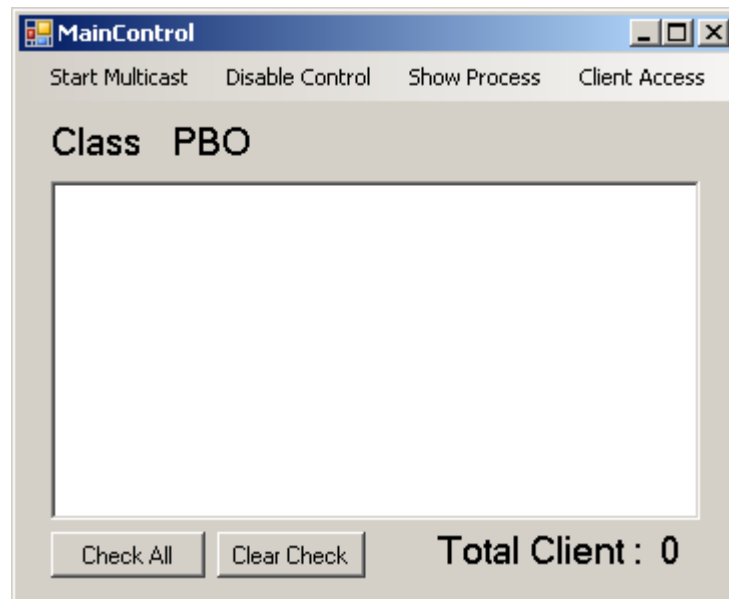
Out Class

Refresh

Gambar 5.3. *Client Join* ke dalam *Class* PBO

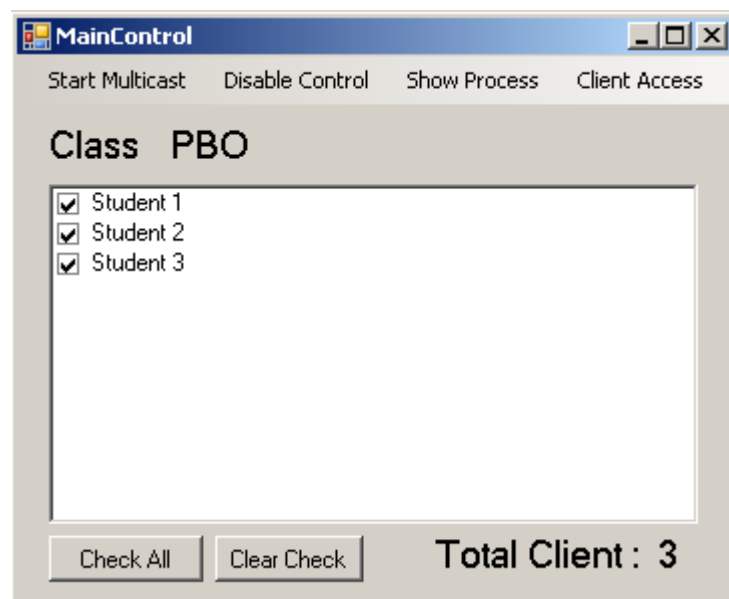
Untuk keluar dari dalam *class*, *client* dapat menekan tombol Out Class dan otomatis *client* keluar dari *class* yang telah diikuti.

Untuk pihak *server*, setelah menekan tombol Create Class akan ditampilkan MainControl Form yang merupakan pusat untuk mengendalikan sebuah *class*. *Server* pada saat ini menunggu adanya *client* yang masuk. Tampilan MainControl Form dapat dilihat pada Gambar 5.4.



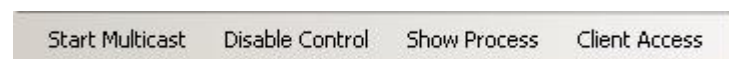
Gambar 5.4. MainControl Form tanpa *Client*

Apabila ada *client* yang telah bergabung, maka tampilan *server* akan berubah seperti pada Gambar 5.5.



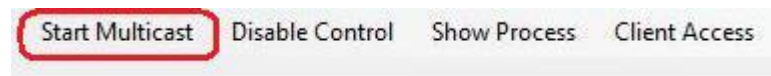
Gambar 5.5. MainControl Form dengan 3 *Client*

Pada MainControl Form terdapat sebuah menu untuk mengendalikan semua kegiatan *class*. Menu tersebut terdiri dari Start Multicast, Disable Control, Show Process dan Client Access. Untuk tampilan menu dapat dilihat pada Gambar 5.6.



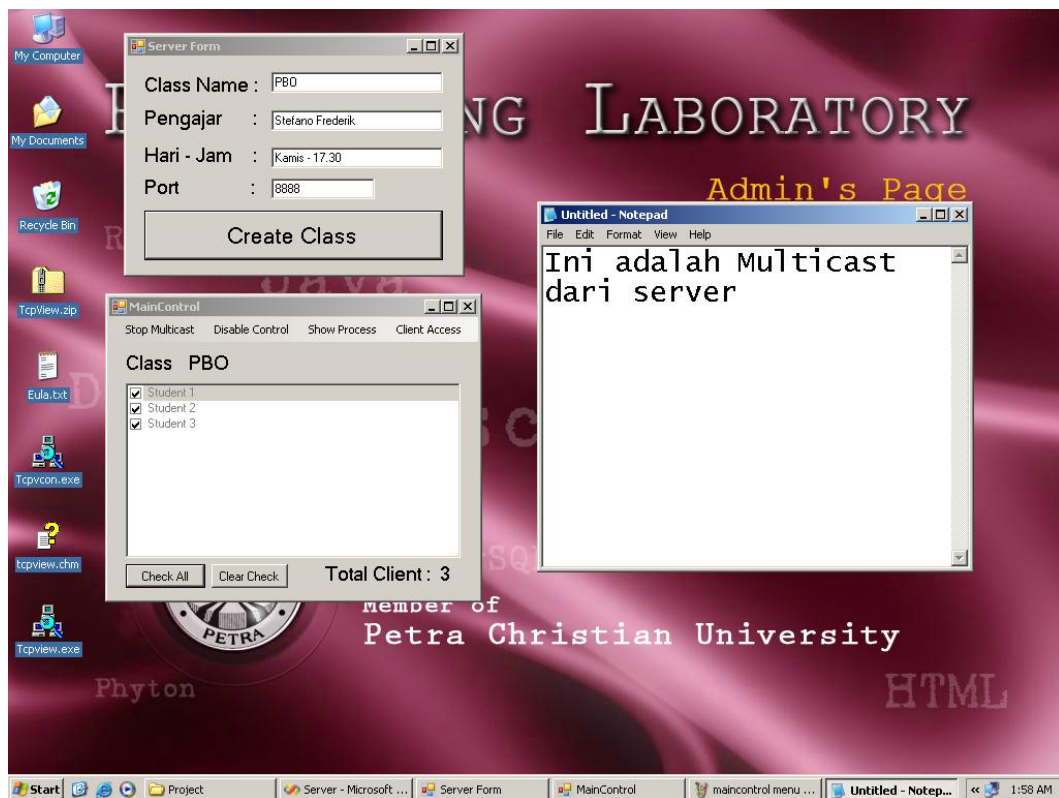
Gambar 5.6. Menu MainControl Form

Apabila *server* ingin melakukan *multicast*, maka *server* dapat langsung menekan pilihan menu Start Multicast.



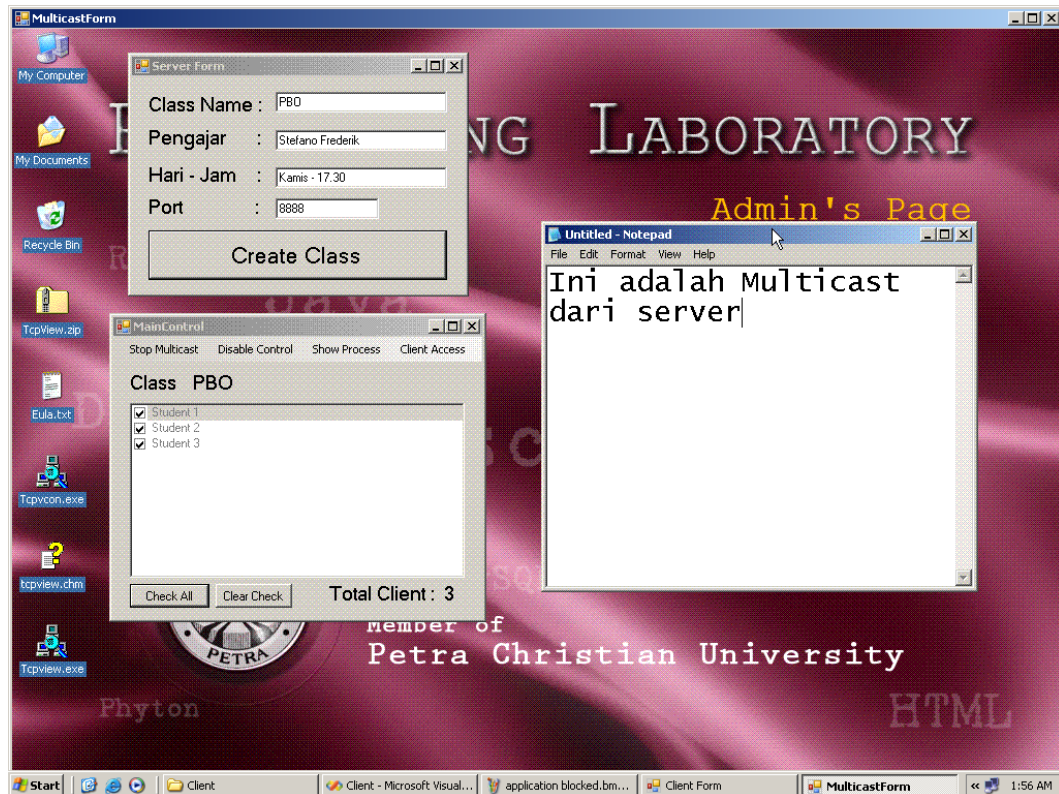
Gambar 5.7. Menu Start Multicast

Saat terjadi *multicast* dari *server*, semua *client* yang telah bergabung akan mendapat tampilan yang sama dengan *server*. Contoh tampilan *server* saat *multicast* bisa dilihat pada Gambar 5.8.



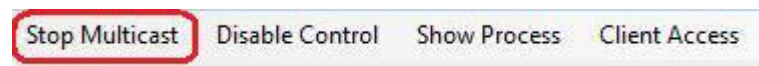
Gambar 5.8. Contoh Tampilan *Server* saat *Multicast*

Client yang menerima gambar dari *server* akan menampilkan pada Multicast Form sesuai dengan gambar tampilan monitor milik *server*. Untuk lebih jelasnya dapat dilihat pada Gambar 5.9.



Gambar 5.9. Multicast Form saat Terjadi *Multicast* dari *Server*

Apabila *server* ingin menghentikan *multicast*, maka *server* dapat memilih pilihan menu Stop Multicast dan secara langsung *client* akan menutup Multicast Form. Tampilan pilihan menu Stop Multicast dapat dilihat pada Gambar 5.10.



Gambar 5.10. Menu Stop Multicast

Menu lainnya yang dapat dipilih adalah Disable Control. Pada menu ini terdapat beberapa menu lagi yaitu Hot Key, Mouse, dan USB. Tampilan pilihan menu Disable Control dapat dilihat pada Gambar 5.11.



Gambar 5.11. Menu Disable Control

Dari pilihan menu Disable Control, *server* dapat memilih lagi antara Hot Key, Mouse, atau USB. Apabila salah satu pilihan telah dipilih, maka akan ada

penanda berupa contreng apabila sedang aktif. Contoh apabila fitur *Hot Key* sedang aktif dapat dilihat pada Gambar 5.12.



Gambar 5.12. Menu Disable Control bagian Hot Key

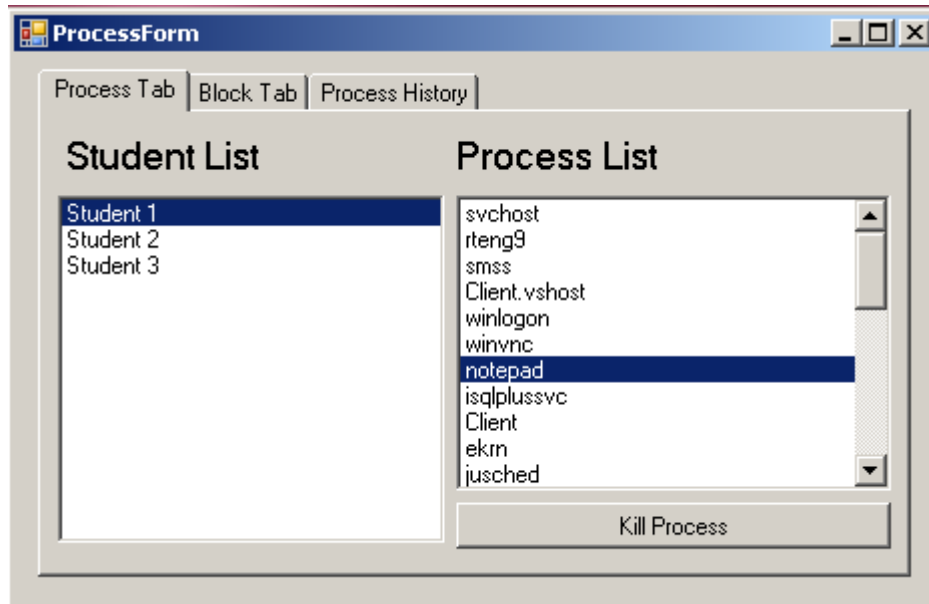
Untuk pilihan menu Mouse dan USB memiliki tampilan yang sama dengan tanda contreng dan apabila ingin melepaskan kembali, maka *server* hanya perlu menekan sekali lagi dan akan kembali ke tampilan sebelumnya seperti pada Gambar 5.4. Efek yang terjadi pada *client* tergantung pada pilihan menu yang dipilih. Apabila pilihan menu Hot Key dipilih, maka semua *hot key* pada *client* tidak dapat dijalankan. Apabila pilihan menu Mouse dipilih, maka fungsi klik kiri pada *mouse* tidak dapat digunakan. Apabila pilihan menu USB dipilih, maka *client* tidak dapat mengenali segala jenis *input* yang berasal dari USB.

Pilihan menu lainnya adalah Show Process. Pilihan menu ini digunakan untuk melihat daftar proses yang sedang dijalankan *client* yang dipilih. Tampilan pilihan menu dapat dilihat pada Gambar 5.13.



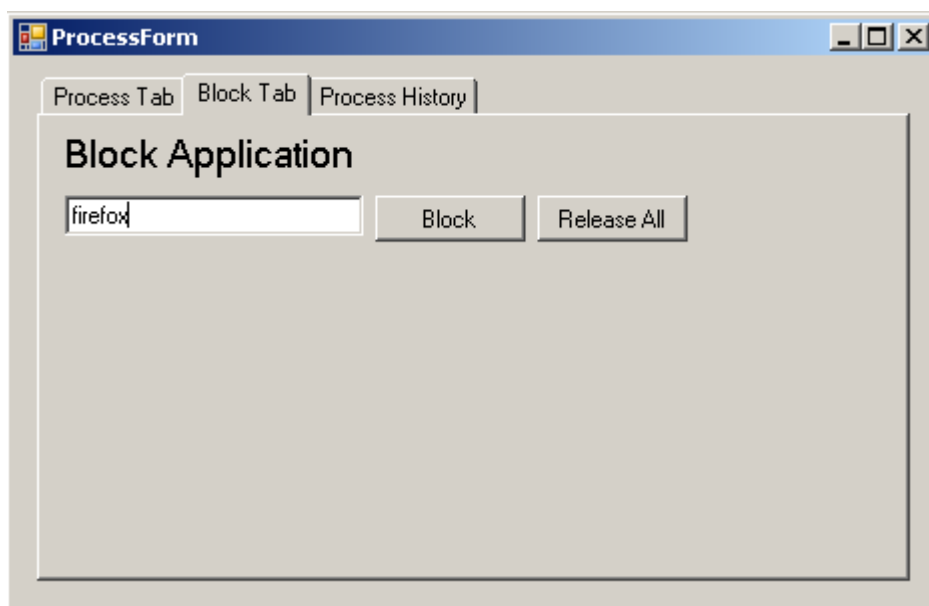
Gambar 5.13. Menu Show Process

Apabila ditekan, maka akan muncul Process Form yang berfungsi untuk mengawasi semua proses yang sedang dilakukan sesuai dengan *client* yang dipilih. Pada Process Form, terdapat 3 *tab* utama, yaitu Process Tab, Block Tab dan Process History Tab. Process Tab digunakan untuk melihat daftar proses yang sedang dijalankan oleh *client*. Terdapat 2 *listbox* utama, yang pertama digunakan untuk menampilkan semua *client* yang ada di dalam *class* dan yang kedua digunakan untuk menampilkan semua proses yang dijalankan sesuai dengan *client* yang dipilih pada *listbox* pertama. Selain itu terdapat tombol Kill Process yang digunakan untuk menutup aplikasi apabila *server* menginginkannya. Contoh *server* saat ingin melihat proses yang dijalankan oleh Student 1 dapat dilihat pada Gambar 5.14.



Gambar 5.14. Tampilan Process Form

Pilihan selanjutnya adalah Block Tab. Block Tab digunakan untuk membatasi proses yang boleh dijalankan oleh semua *client*. Pada Block Tab terdapat *textbox* untuk memasukkan nama proses yang ingin dibatasi dan bersifat *case sensitive*, tombol Block untuk mengirim perintah *block application* kepada semua *client* sesuai dengan *input* dari *textbox* dan tombol Release All untuk memperbolehkan kembali semua aplikasi untuk dijalankan. Contoh *server* ingin membatasi aplikasi firefox.



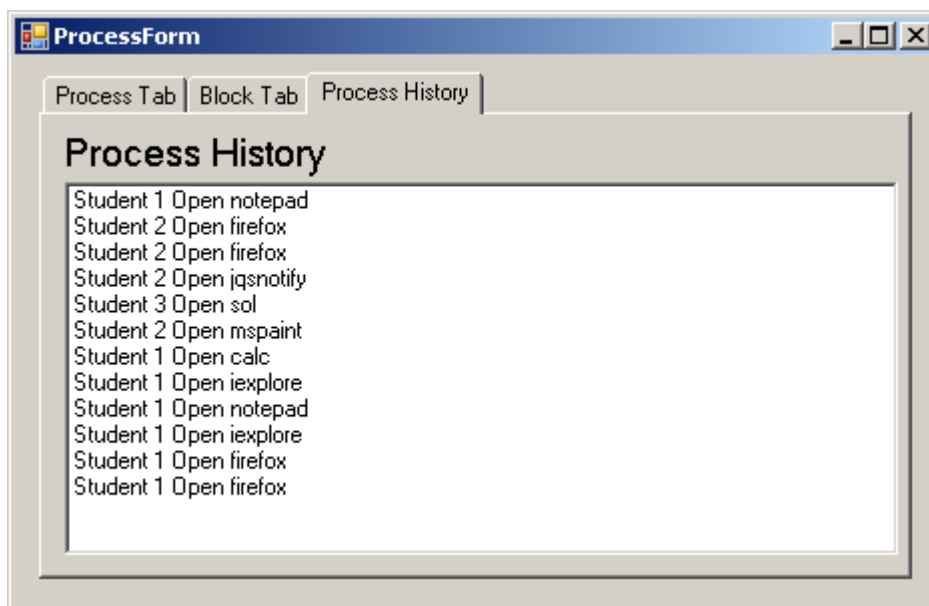
Gambar 5.15. Tampilan Block Application Tab

Efek yang terjadi pada *client* saat suatu aplikasi dibatasi adalah dengan menampilkan sebuah *message box* yang memberitahukan bahwa aplikasi tersebut tidak dapat dijalankan. Untuk lebih jelasnya dapat dilihat pada Gambar 5.16.



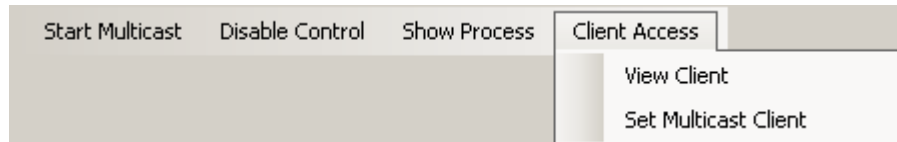
Gambar 5.16. *Message box* Sebuah Aplikasi Dibatasi

Pilihan selanjutnya adalah Process History. Process History digunakan untuk melihat aplikasi apa saja yang telah dijalankan oleh *client*. Hal ini dilakukan untuk memantau keadaan *client*. Apabila *client* membuka sebuah aplikasi, maka saat itu juga *client* akan mengirimkan pemberitahuan kepada *server* aplikasi apa yang dijalankan. Untuk tampilannya dapat dilihat pada Gambar 5.17.



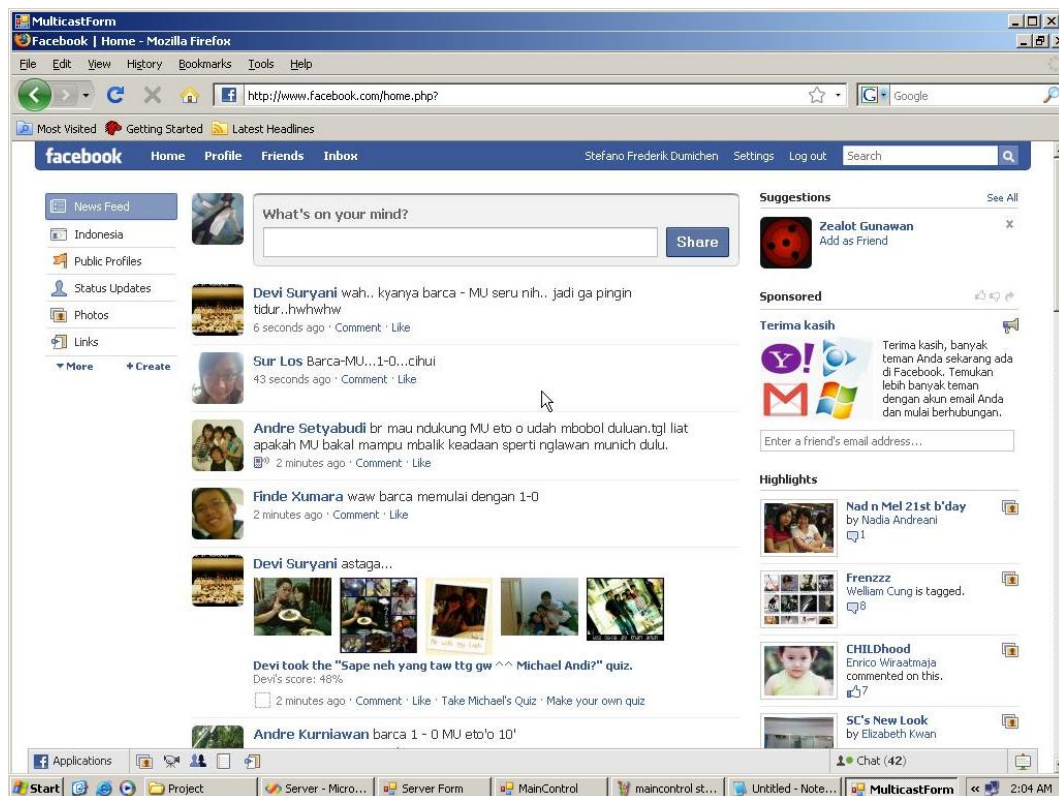
Gambar 5.17. Tampilan Process History *Client*

Pilihan menu yang terakhir adalah Client Access. Pada menu ini, terdapat dua pilihan menu lagi yaitu View Client dan Set Multicast Client. Pilihan menu View Client digunakan untuk melihat tampilan 1 *client* sesuai dengan pilihan sedangkan untuk Set Multicast Client digunakan untuk meminta *client* yang dipilih untuk melakukan *multicast*. Untuk menu Client Access dapat dilihat pada Gambar 5.18.



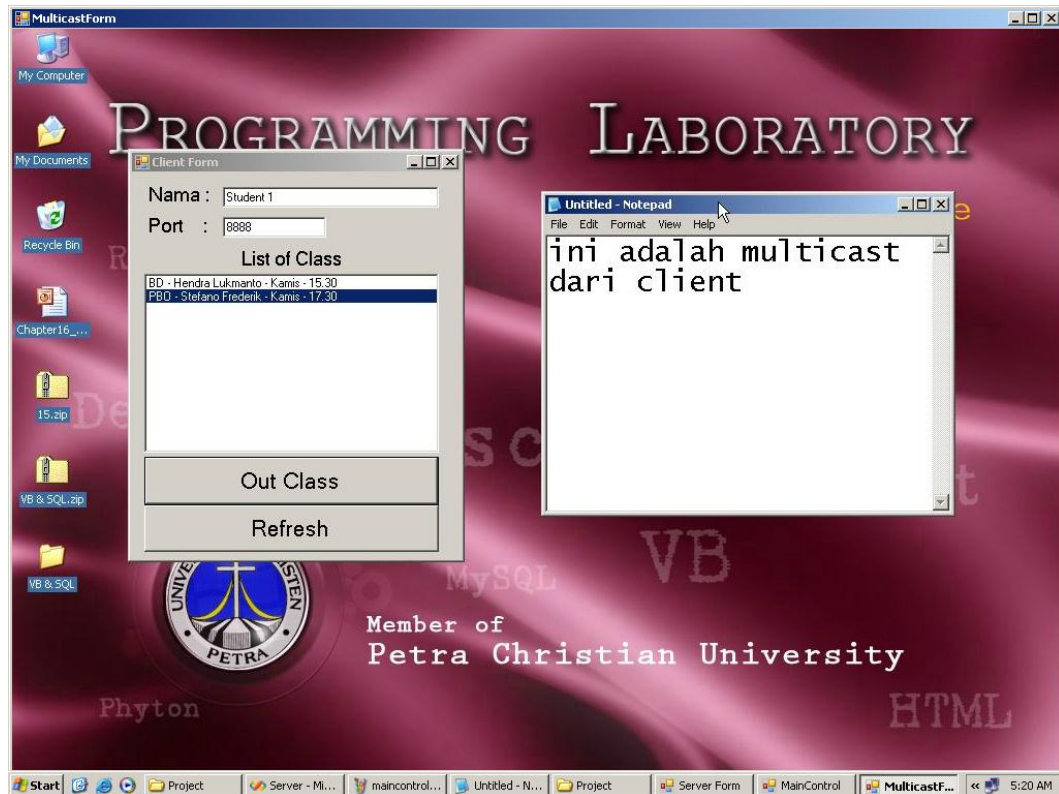
Gambar 5.18. Menu Client Access

Jika pilihan menu View Client dipilih, maka *client* yang telah dipilih akan langsung mengirimkan data gambar kepada *server* agar *server* dapat melakukan pengawasan. Data gambar yang dikirim oleh *client* akan ditampilkan oleh *server* pada Multicast Form. Batasan untuk View Client adalah tidak bisa dijalankan bersamaan dengan Start Multicast dan Set Multicast Client. Untuk lebih jelasnya, dapat dilihat pada Gambar 5.19.



Gambar 5.19. Menu Client Access View Client

Pilihan menu kedua pada Client Access adalah Set Multicast Client. Fungsi ini hampir sama dengan Start Multicast tetapi yang melakukan *multicast* bukanlah *server* sendiri melainkan *client* yang dipilih oleh *server*. Batasan untuk Set Multicast Client adalah tidak bisa dijalankan bersamaan dengan Start Multicast dan View Client. Data gambar yang dikirimkan oleh *client* akan ditampilkan pada Multicast Form. Untuk lebih jelasnya dapat dilihat pada Gambar 5.20.



Gambar 5.20. Set Multicast Client

5.2. Pengujian Program

Pengujian aplikasi dilakukan dengan membandingkan konsumsi *bandwidth*, *CPU usage* pada setiap jumlah *client* dan *server* yang bervariasi, perubahan ukuran resolusi monitor *server* maupun *client*.

5.2.1. Pengujian Semua Fitur *Client* pada 1 *server*

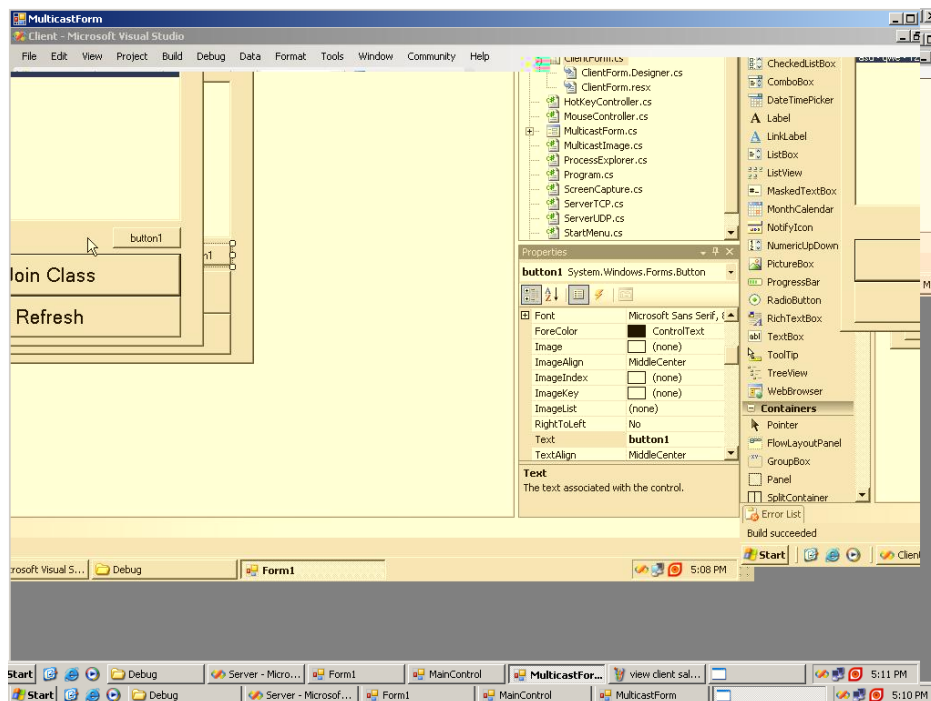
Pengujian dilakukan dengan menggunakan jumlah *client* yang berubah-ubah. Yang diujikan adalah seberapa kuat sebuah *server* untuk melayani sejumlah *client* tertentu. Semua fitur akan diujikan antara lain *multicast*, *block keyboard*, *block mouse*, dan *block application*. Jumlah *client* yang akan diujikan adalah 5,10,15,20,25,30,40.

Tabel 5.1. Pengujian Seluruh Fitur pada Beberapa *Client*

Jumlah <i>client</i>	<i>Multicast*</i>	<i>Block keyboard**</i>	<i>Block mouse**</i>	<i>Block application</i>	<i>View client*</i>	<i>Multicast client*</i>
5	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
10	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
15	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
20	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
25	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
30	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses
40	Sukses	Sukses	Sukses	Sukses	Sukses	Sukses

Keterangan :

- * pada saat *multicast* terkadang gambar pada *client* berkedip atau dengan kata lain gambar yang diterima rusak pada suatu waktu yang tidak tentu. Hal ini disebabkan karena memakai IP *multicast* yang merupakan protocol *connectionless* sehingga ada *packet* data yang hilang. Contoh gambar yang rusak saat *multicast* dapat dilihat pada Gambar 5.21.



Gambar 5.21. Data Gambar Salah saat Start Multicast, View Client, dan Multicast Client

- ** pada saat *block keyboard* dan *block mouse* berlaku juga untuk *keyboard* dan *mouse* yang memakai USB. Apabila saat terjadi *block keyboard* dan *block mouse* tiba-tiba ditancapkan *keyboard* atau *mouse* yang menggunakan USB, maka *keyboard* ataupun *mouse* tersebut tidak bisa digunakan pula.

Dari pengujian di atas, terdapat *error* pada saat terjadi *multicast*, yaitu terjadi gambar yang rusak atau gagal. Untuk fitur *block keyboard* dan *block mouse* berjalan dengan baik untuk semua jumlah *client*. Untuk fitur *view client* dan *multicast client*, terjadi masalah yang sama dengan *multicast* karena memakai cara yang sama sehingga mengakibatkan gambar berkedip pada suatu waktu yang tidak tentu.

5.2.2. Pengujian *Delay* Pada *Client* Saat *Multicast*

Pengujian ini merupakan penghitungan waktu yang digunakan saat terjadi *multicast*. Beberapa hal yang diujikan adalah tipe format gambar dan resolusi monitor.

5.2.2.1. Berdasarkan Tipe Format Gambar

Untuk pengujian *delay* saat *multicast*, yang diujikan adalah *input* dari gambar yang akan dikirim. Tipe format gambar yang akan diujikan adalah gif, jpeg, bmp, dan png. Pengujian dilakukan dengan cara menulis sebuah huruf pada aplikasi notepad dan menghitung berapa lama huruf tersebut keluar di layar *client* dengan menggunakan *stopwatch* dalam 10 kali percobaan. Ukuran waktu yang digunakan adalah satuan detik. Resolusi monitor yang dipakai adalah 1024 x 768.

Tabel 5.2. Pengujian *Delay* Berdasarkan Tipe Format Gambar

Tipe format	GIF	JPEG	BMP	PNG
Percobaan 1	00.88	01.48	11.22	00.90
Percobaan 2	01.17	01.20	11.53	01.39
Percobaan 3	01.18	00.98	11.86	01.22
Percobaan 4	00.90	01.27	10.98	01.30
Percobaan 5	01.18	01.01	09.29	00.95

Tabel 5.2. Pengujian *Delay* Berdasarkan Tipe Format Gambar (sambungan)

Percobaan 6	00.88	01.22	11.96	01.19
Percobaan 7	01.18	01.28	12.44	00.92
Percobaan 8	00.85	01.20	09.16	01.20
Percobaan 9	00.94	01.18	11.85	01.05
Percobaan 10	01.01	00.97	12.38	00.85
Rata – rata :	1.017	1.179	11.267	1.097

Dari hasil pengujian di atas, terlihat bahwa *delay* terbesar ada pada saat tipe format BMP dan *delay* terkecil adalah tipe format GIF.

5.2.2.2. Berdasarkan Resolusi Monitor Server

Proses pengujian konsumsi *bandwidth* dilakukan dengan mengubah ukuran resolusi monitor server. Ukuran resolusi monitor yang akan diujikan adalah 800x600, 1024x768, 1152x864, 1280x768 dan 1440x900. Resolusi monitor *client* yang digunakan adalah 800 x 600. Jumlah *client* yang terhubung tidak diperhitungkan karena sesuai pengujian sebelumnya bahwa jumlah *client* tidak mempengaruhi *bandwidth*.

Tabel 5.3. Pengujian *Delay* Berdasarkan Resolusi Monitor

Ukuran Resolusi	800 x 600	1024 x 768	1152 x 864	1280 x 768	1440 x 900
Percobaan 1	00.64	00.88	01.02	01.20	01.33
Percobaan 2	00.65	00.72	00.97	01.01	01.19
Percobaan 3	00.65	00.91	00.95	01.17	01.20
Percobaan 4	00.69	00.82	01.18	00.99	01.24
Percobaan 5	00.70	00.96	01.25	00.85	01.24
Percobaan 6	00.59	00.79	01.18	01.18	01.31
Percobaan 7	00.63	00.74	00.94	01.19	01.47
Percobaan 8	00.59	00.49	01.12	01.18	01.32
Percobaan 9	00.65	00.72	01.20	00.95	01.39
Percobaan 10	00.64	00.76	00.86	01.16	00.99
Rata – rata :	0.643	0.779	1.067	1.088	1.268

Dari Tabel 5.3 di atas, dapat disimpulkan bahwa perubahan resolusi monitor mempengaruhi *delay*. Makin besar ukuran monitor yang dipakai maka makin *lama* *delay* yang ada. Hal ini disebabkan karena waktu yang diperlukan untuk membaca data gambar makin lama apabila data gambar makin besar.

5.2.3. Pengujian Konsumsi *Bandwidth*

Pengujian ini merupakan pengecekan konsumsi *bandwidth* yang dipakai oleh *server* selama terjadi *multicast*. Beberapa hal yang diujikan adalah jumlah *client* yang terhubung dalam 1 *server* dan tipe format gambar.

5.2.3.1. Berdasarkan Jumlah *Client* pada 1 *Server*

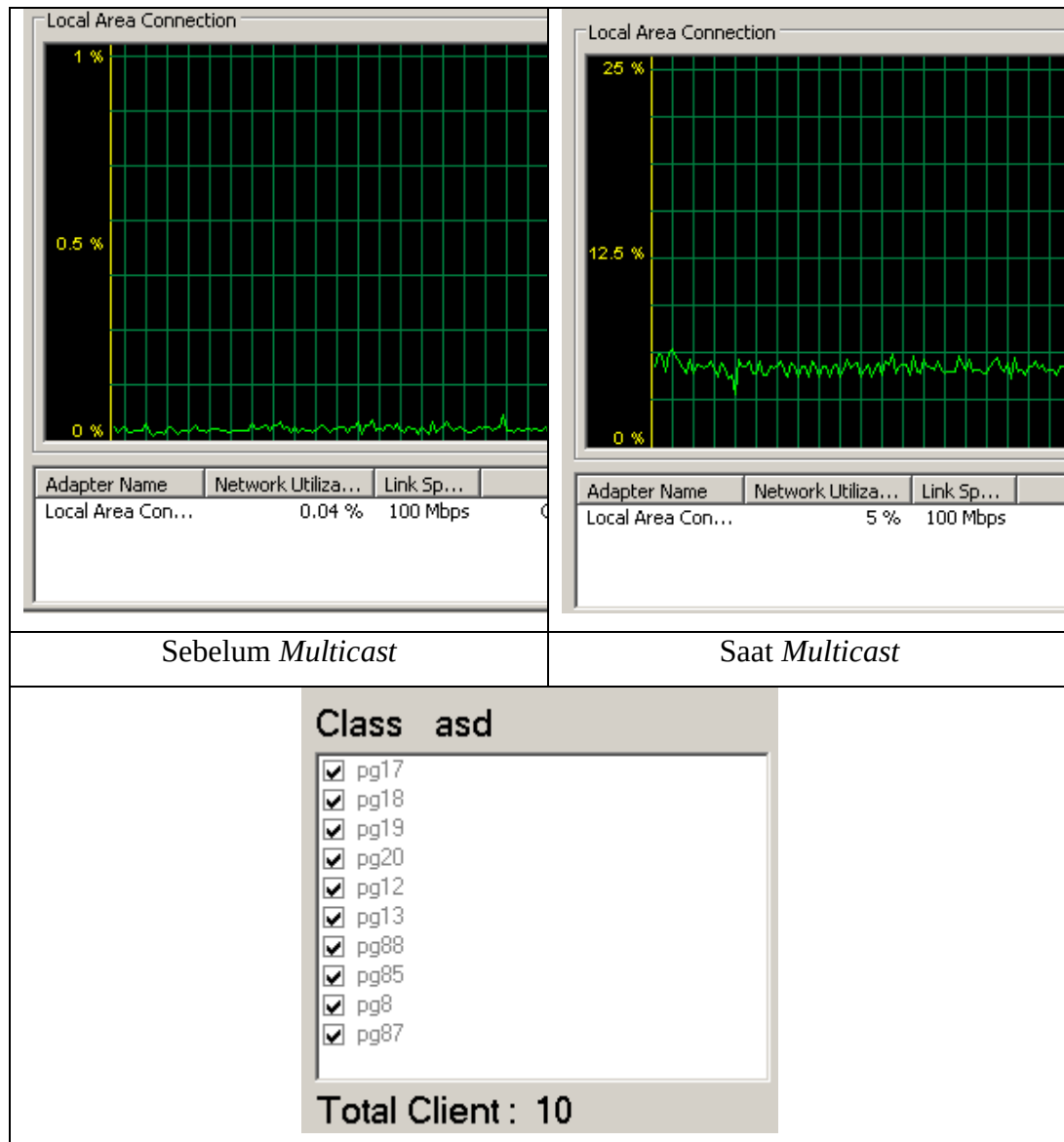
Proses pengujian konsumsi *bandwidth* dilakukan dengan menggunakan jumlah *client* yang berbeda-beda pada 1 *server*. Pengujian pemakaian *bandwidth* dilakukan pada saat *server* melakukan *multicast* ke semua *client* yang terhubung. Konsumsi *bandwidth* yang diuji adalah *bandwidth* yang dipakai oleh *server* maupun *client* pada saat *multicast*. Jumlah *client* yang akan diujikan adalah 5, 10, 15, 20, 25, 30, 40.

Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi

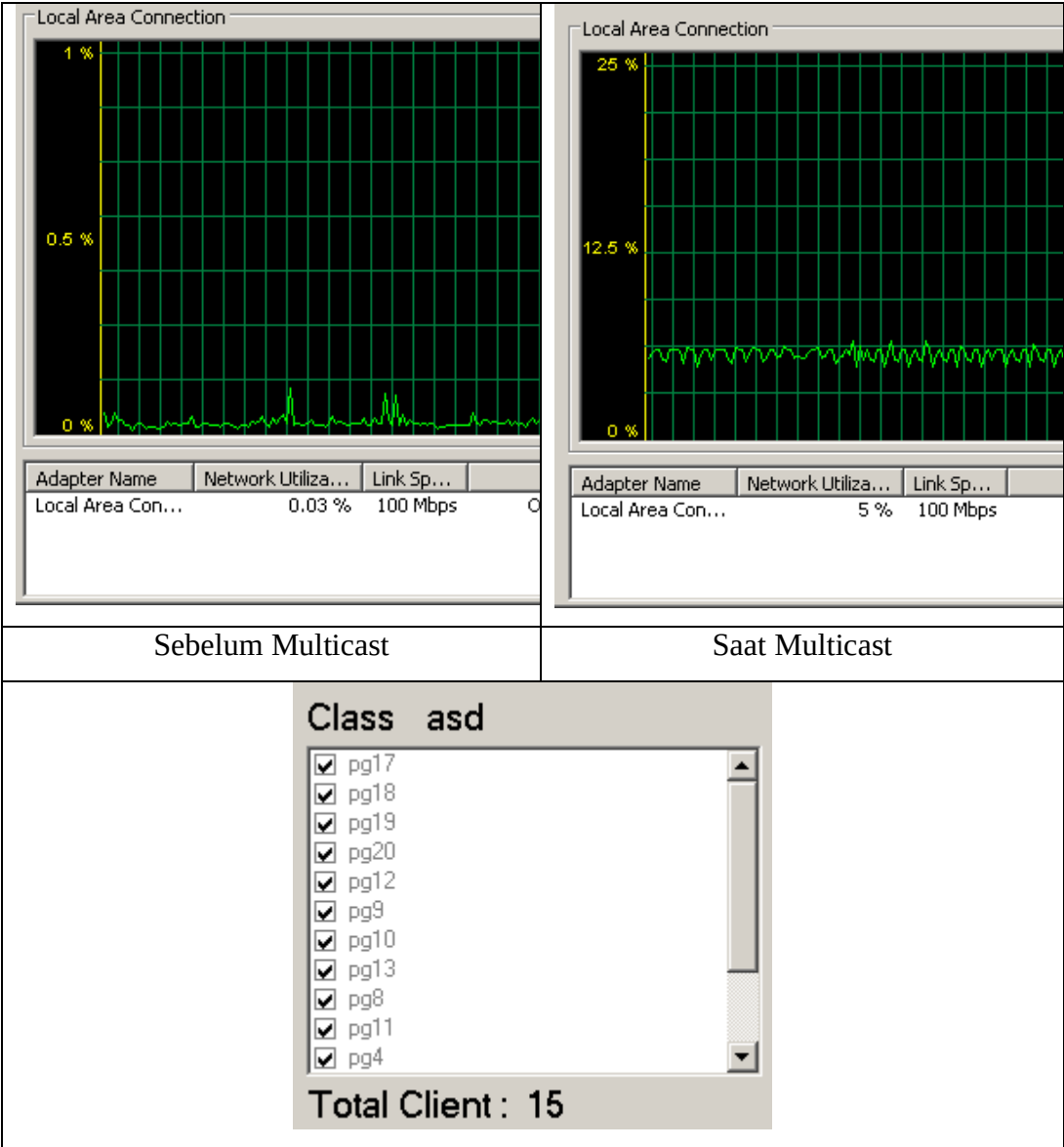
Class	Testing Class
☒	pg17
☒	pg18
☒	pg12
☒	pg19
☒	pg20

Total Client : 5

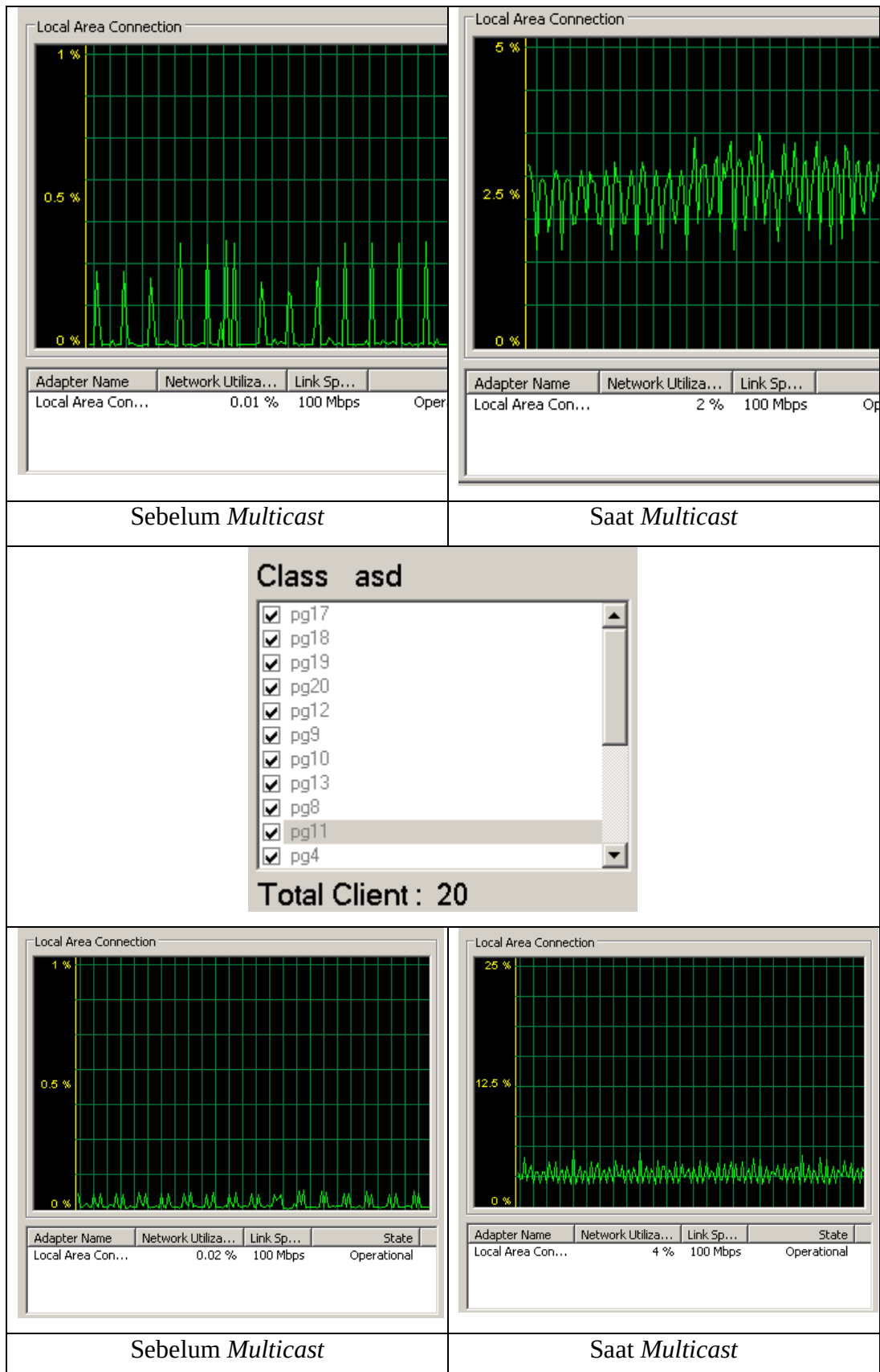
Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi (sambungan)



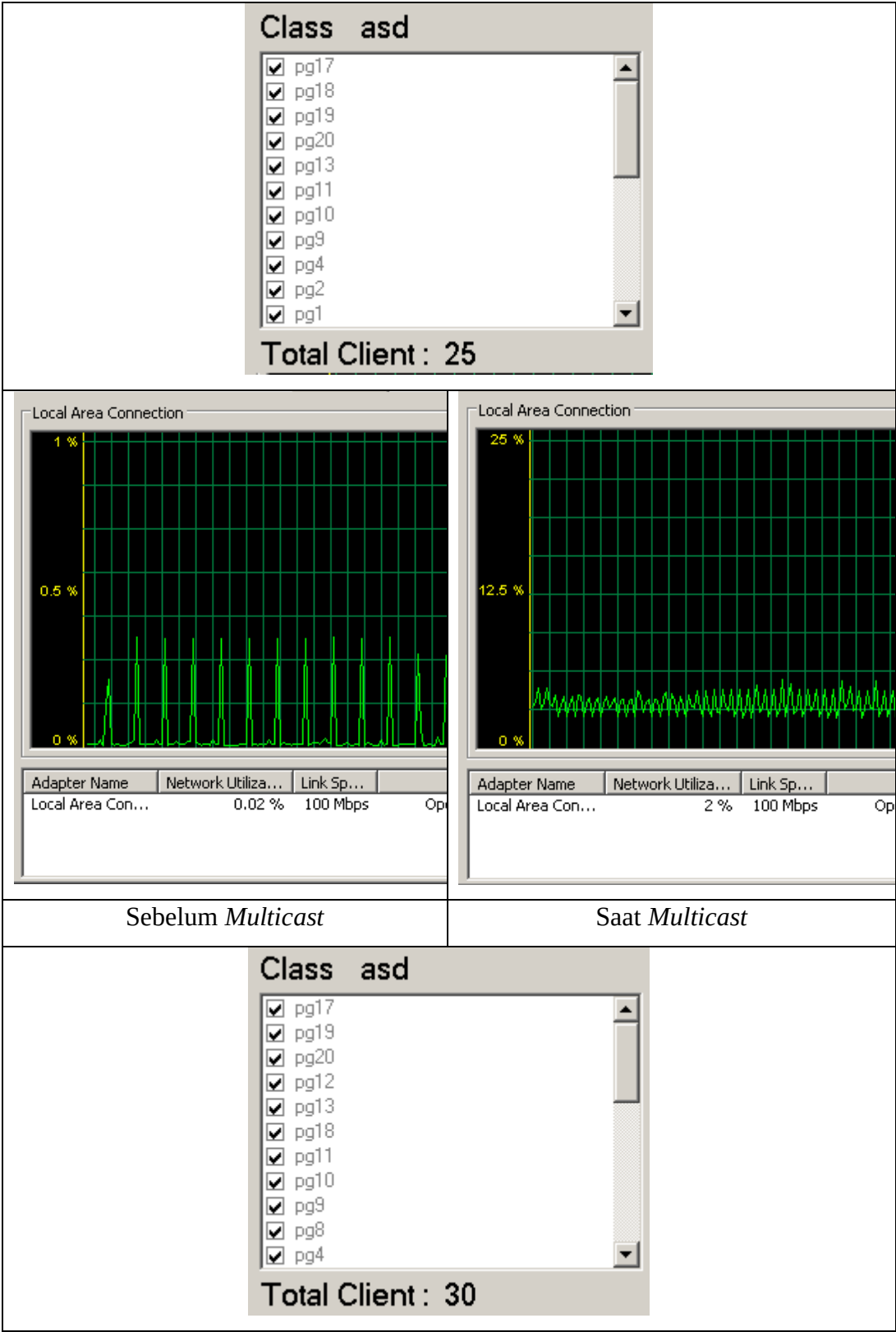
Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi (sambungan)



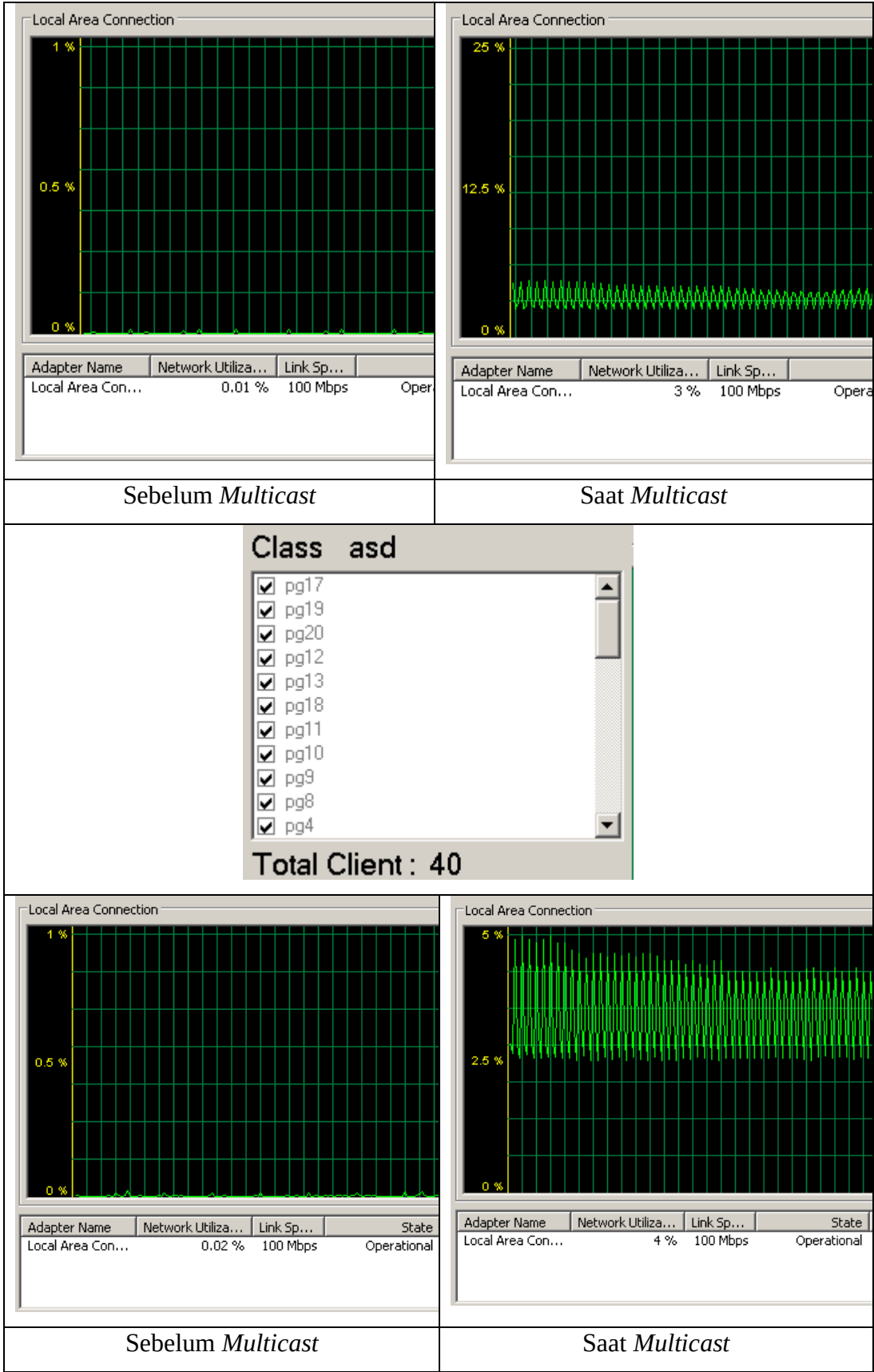
Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi (sambungan)



Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi (sambungan)



Tabel 5.4. Penggunaan *Bandwidth* dengan Jumlah *Client* Bervariasi (sambungan)

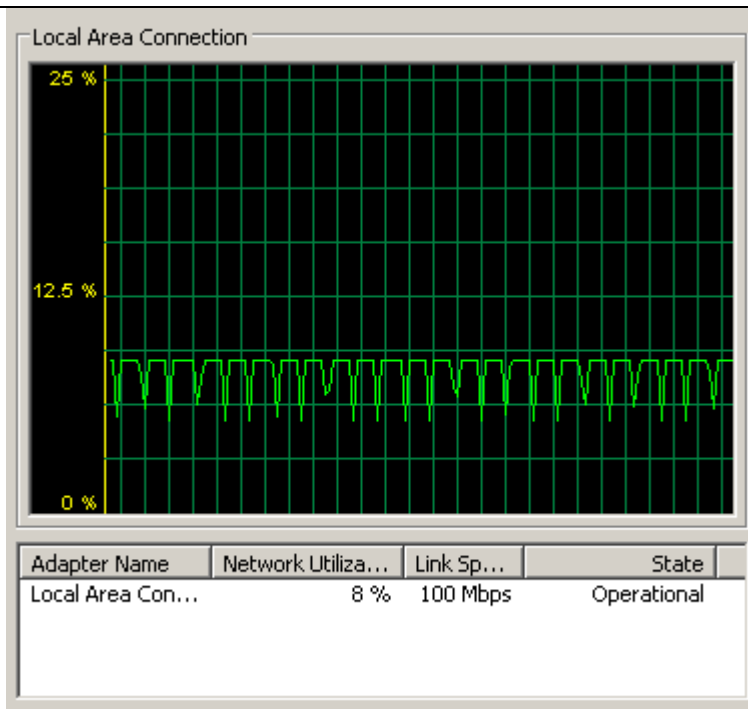


Dari hasil Tabel 5.4, dapat disimpulkan bahwa penambahan jumlah *client* tidak mempengaruhi *bandwidth* yang digunakan *server* untuk melakukan *multicast*. Pengujian tiap jumlah *client* dilakukan dalam waktu kurang lebih 5 menit. Pemakaian *bandwidth* sebelum terjadi *multicast* adalah sekitar 0.01% sampai dengan 0.02% yang berarti 1 Kb sampai dengan 2 Kb. Saat terjadi *multicast*, *server* mengirim data berupa gambar secara terus menerus sehingga terjadi peningkatan dalam jumlah *bandwidth* yaitu sekitar 2% sampai dengan 5% yang berarti sekitar 2 Mb sampai dengan 5 Mb.

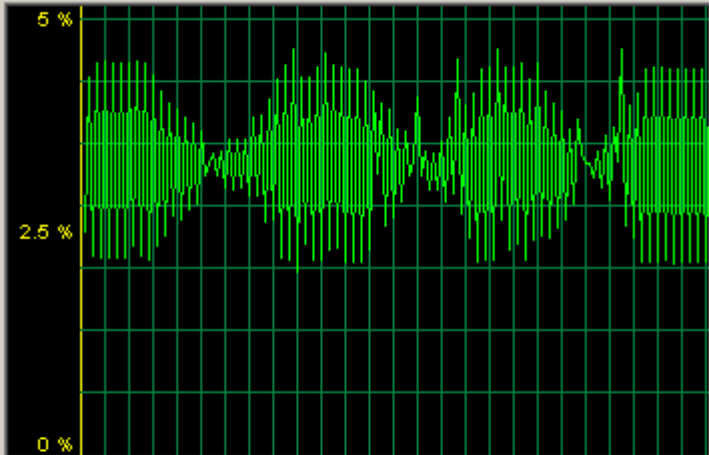
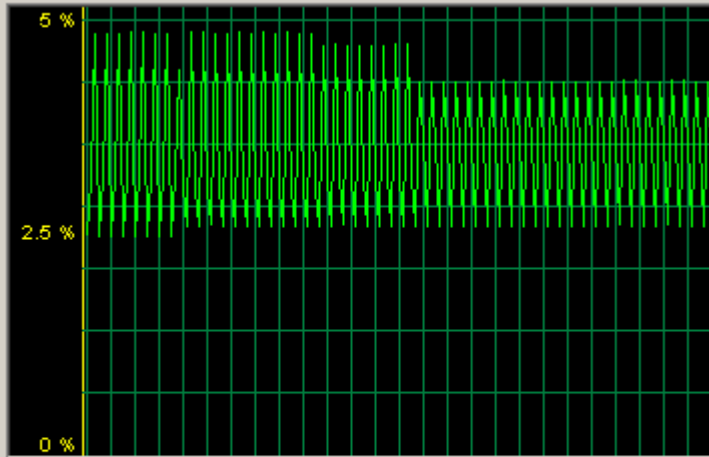
5.2.3.2. Berdasarkan Tipe Format Gambar

Proses pengujian konsumsi *bandwidth* dilakukan dengan mengubah tipe gambar yang dikirim kepada *client*. Tipe gambar yang dipakai adalah bmp, jpeg, gif, dan png. Jumlah *client* yang akan digunakan tidak diperhitungkan karena sesuai pengujian sebelumnya bahwa jumlah *client* tidak berpengaruh pada *bandwidth* yang dipakai. Penggunaan *Bandwidth* berdasarkan tipe format gambar

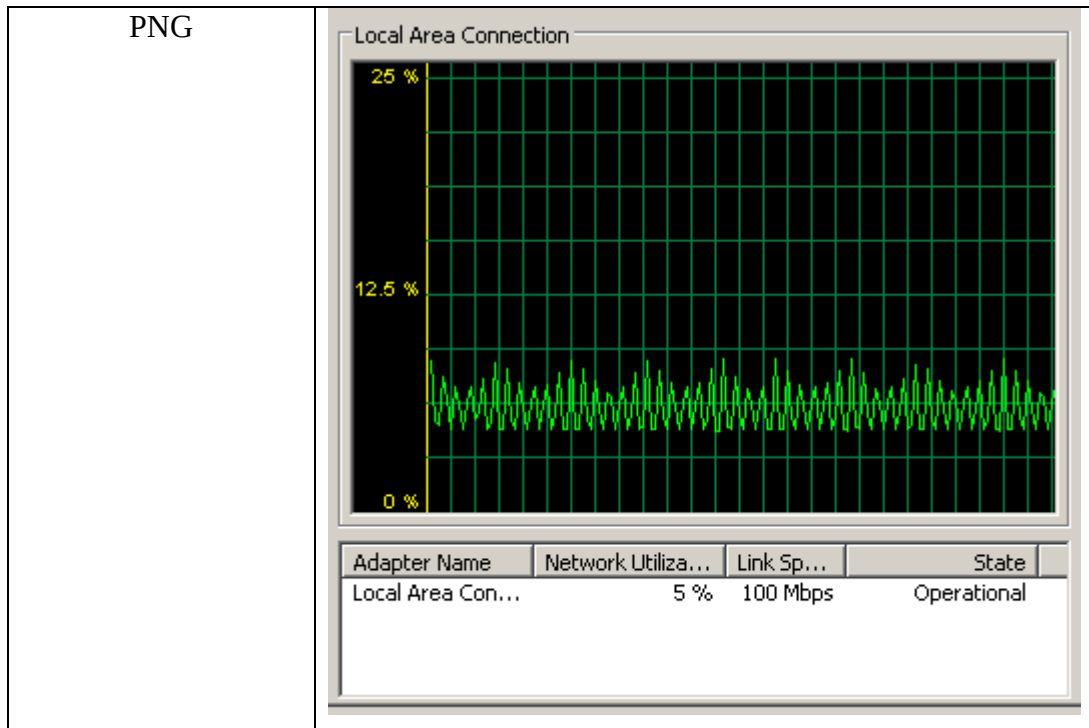
Tabel 5.5. Pengujian Konsumsi *Bandwidth* Berdasarkan Tipe Format Gambar

Tipe Format	Saat <i>Multicast</i>
BMP	

Tabel 5.5. Pengujian Konsumsi *Bandwidth* Berdasarkan Tipe Format Gambar
(sambungan)

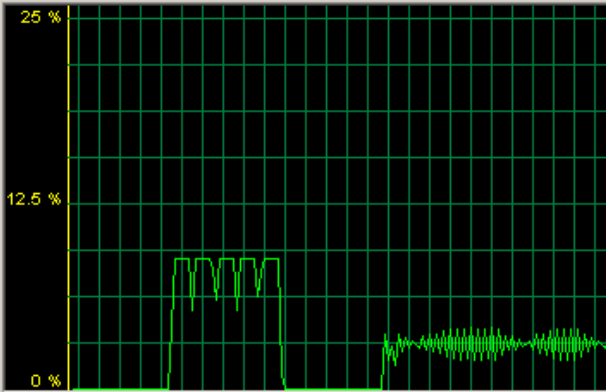
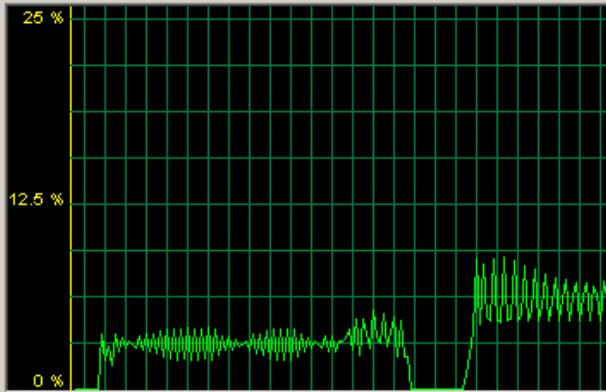
JPEG	Local Area Connection  <table><tr><th>Adapter Name</th><th>Network Utiliza...</th><th>Link Sp...</th><th>State</th></tr><tr><td>Local Area Con...</td><td>4 %</td><td>100 Mbps</td><td>Operational</td></tr></table>	Adapter Name	Network Utiliza...	Link Sp...	State	Local Area Con...	4 %	100 Mbps	Operational
Adapter Name	Network Utiliza...	Link Sp...	State						
Local Area Con...	4 %	100 Mbps	Operational						
GIF	Local Area Connection  <table><tr><th>Adapter Name</th><th>Network Utiliza...</th><th>Link Sp...</th><th>State</th></tr><tr><td>Local Area Con...</td><td>4 %</td><td>100 Mbps</td><td>Operational</td></tr></table>	Adapter Name	Network Utiliza...	Link Sp...	State	Local Area Con...	4 %	100 Mbps	Operational
Adapter Name	Network Utiliza...	Link Sp...	State						
Local Area Con...	4 %	100 Mbps	Operational						

Tabel 5.5. Pengujian Konsumsi *Bandwidth* Berdasarkan Tipe Format Gambar
(sambungan)

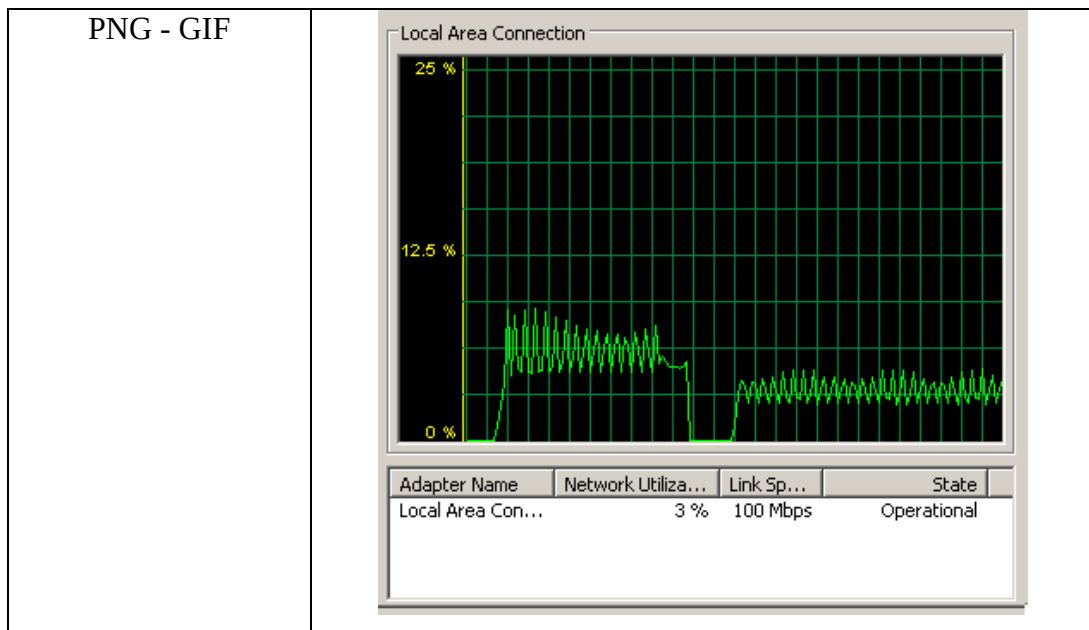


Dari hasil Tabel 5.5, dapat disimpulkan bahwa *bandwidth* terbesar ada pada saat tipe format gambar BMP yaitu sekitar 8% atau dengan kata lain berkisar 8 Mb. Untuk tipe format JPEG, GIF dan PNG memakai sekitar 4% - 5% atau sekitar 4 Mb – 5 Mb saja.

Tabel 5.6. Perbandingan Penggunaan *Bandwidth* Terhadap Tipe Format Gambar

Tipe Format	Saat <i>Multicast</i>								
BMP - JPEG	Local Area Connection  <table><thead><tr><th>Adapter Name</th><th>Network Utiliza...</th><th>Link Sp...</th><th>State</th></tr></thead><tbody><tr><td>Local Area Con...</td><td>3 %</td><td>100 Mbps</td><td>Operational</td></tr></tbody></table>	Adapter Name	Network Utiliza...	Link Sp...	State	Local Area Con...	3 %	100 Mbps	Operational
Adapter Name	Network Utiliza...	Link Sp...	State						
Local Area Con...	3 %	100 Mbps	Operational						
JPEG - PNG	Local Area Connection  <table><thead><tr><th>Adapter Name</th><th>Network Utiliza...</th><th>Link Sp...</th><th>State</th></tr></thead><tbody><tr><td>Local Area Con...</td><td>5 %</td><td>100 Mbps</td><td>Operational</td></tr></tbody></table>	Adapter Name	Network Utiliza...	Link Sp...	State	Local Area Con...	5 %	100 Mbps	Operational
Adapter Name	Network Utiliza...	Link Sp...	State						
Local Area Con...	5 %	100 Mbps	Operational						

Tabel 5.6. Perbandingan Penggunaan *Bandwidth* Terhadap Tipe Format Gambar
(sambungan)



Dari Tabel 5.6 diatas, dapat disimpulkan bahwa pemakaian *bandwidth* terbesar adalah ketika gambar dengan tipe format BMP. Pemakaian terbesar kedua adalah tipe format PNG, sedangkan untuk JPEG dan GIF tidak berbeda jauh pemakaian *bandwidth*-nya.

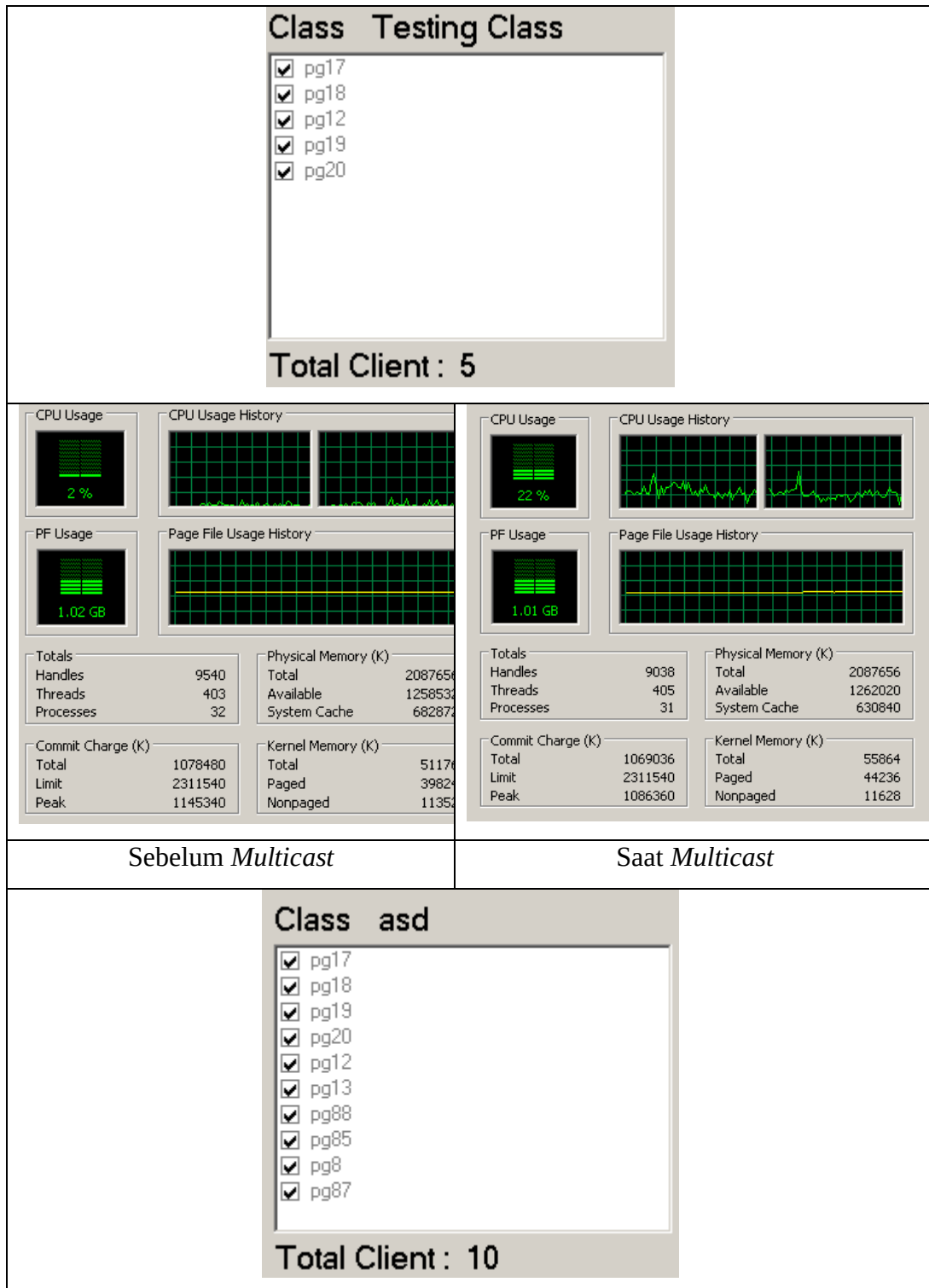
5.2.4. Pengujian Penggunaan CPU Usage

Pengujian ini merupakan pengecekan pemakaian CPU Usage pada *server* pada saat terjadi *multicast*. Hal-hal yang diujikan adalah jumlah *client*.

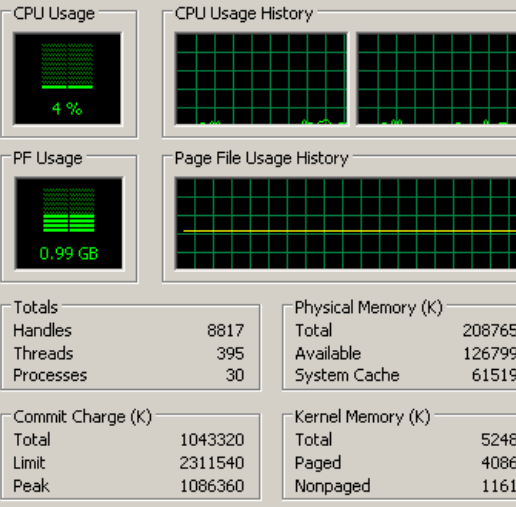
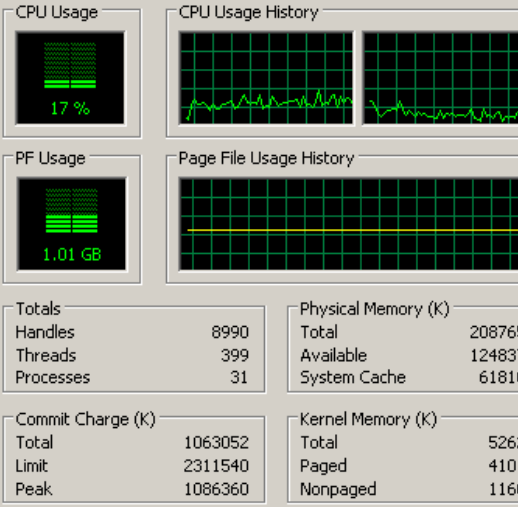
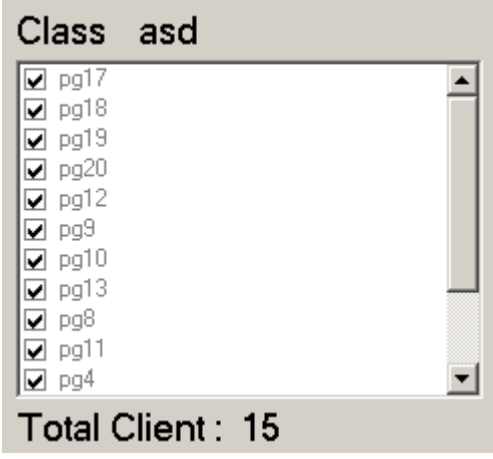
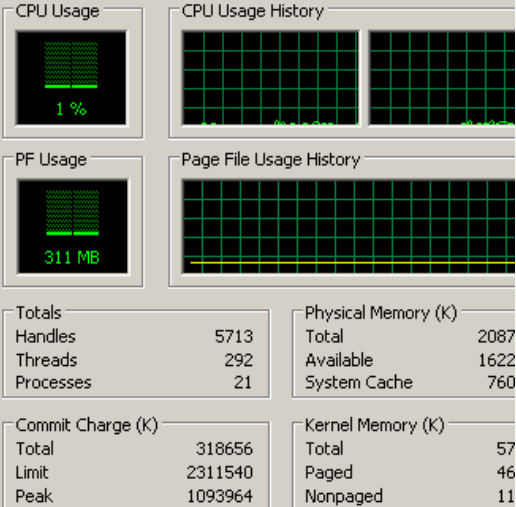
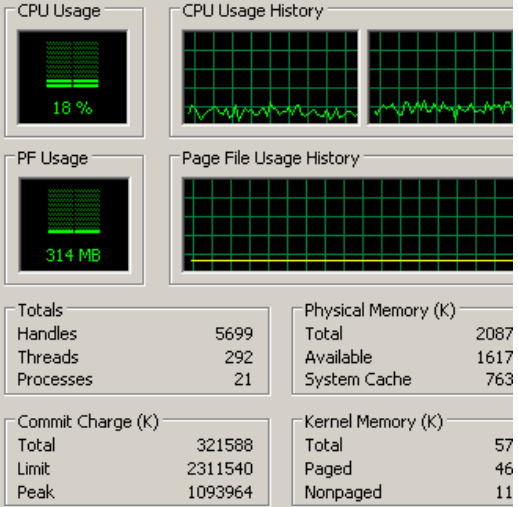
5.2.4.1. Berdasarkan Jumlah Client pada 1 Server

Proses pengujian penggunaan CPU usage dilakukan dengan jumlah *client* yang berbeda-beda pada 1 *server*. Pengujian penggunaan CPU usage dilakukan saat terjadi *multicast* kepada semua *client* yang terhubung. Pemakaian CPU usage yang diuji adalah CPU usage yang dipakai oleh *server* dan *client*. Jumlah *client* yang akan diujikan adalah 5,10,15,20,25,30,40.

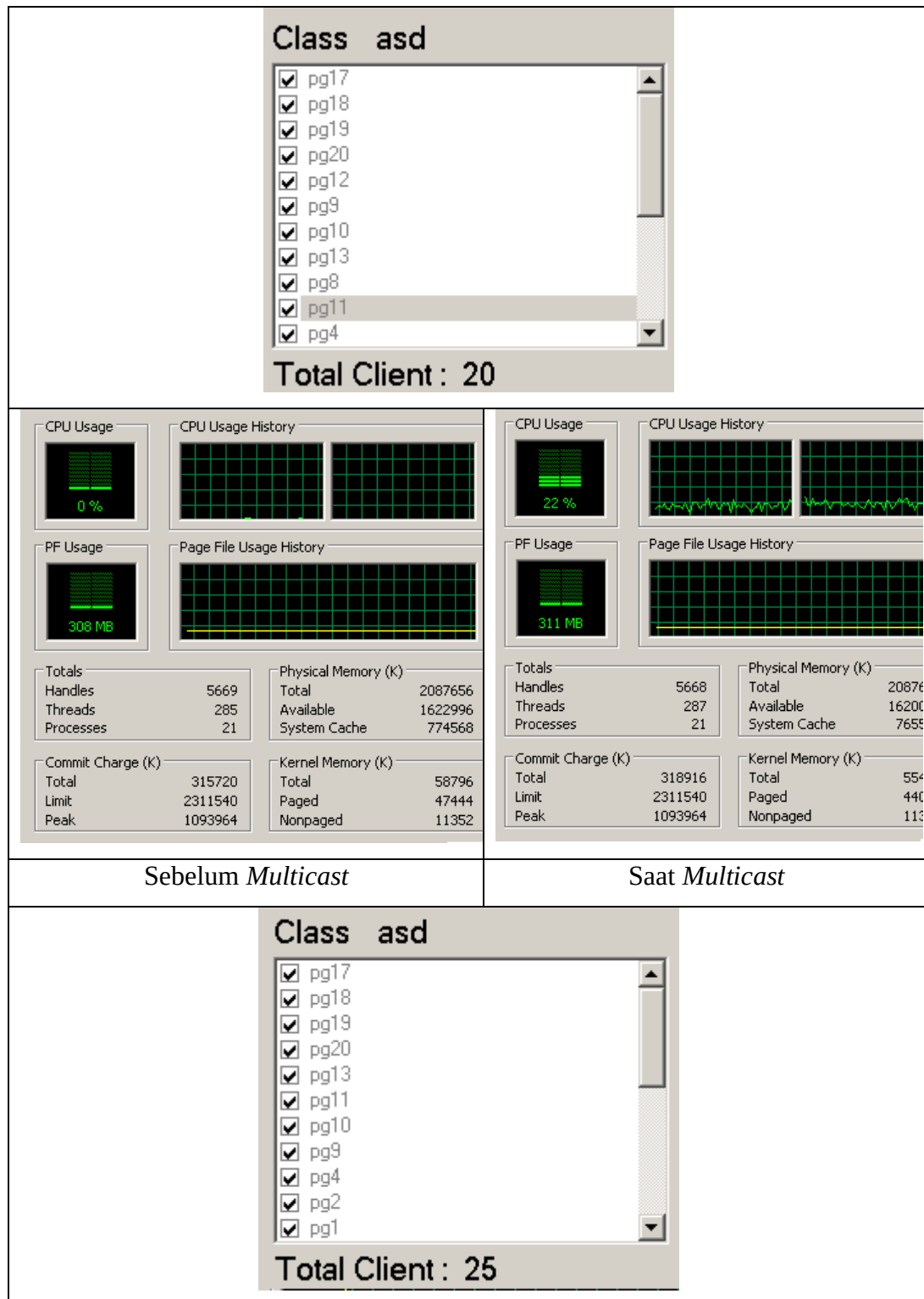
Tabel 5.7. Pengujian CPU Usage pada 1 Server



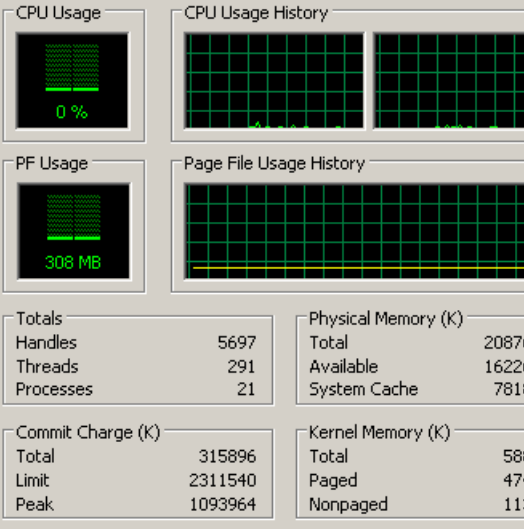
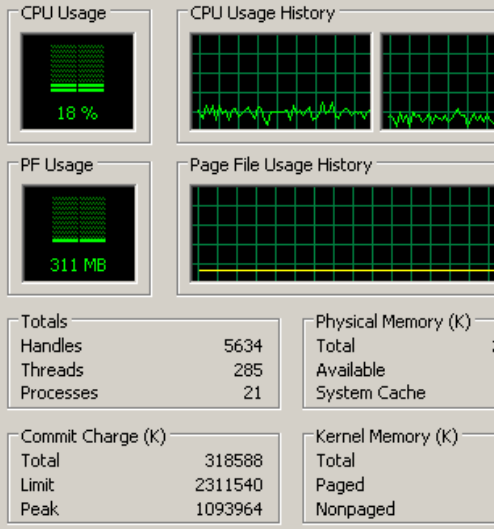
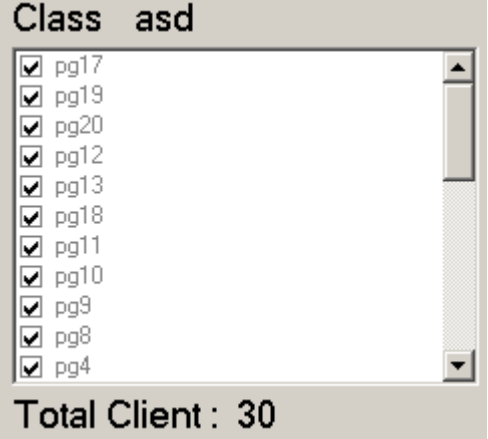
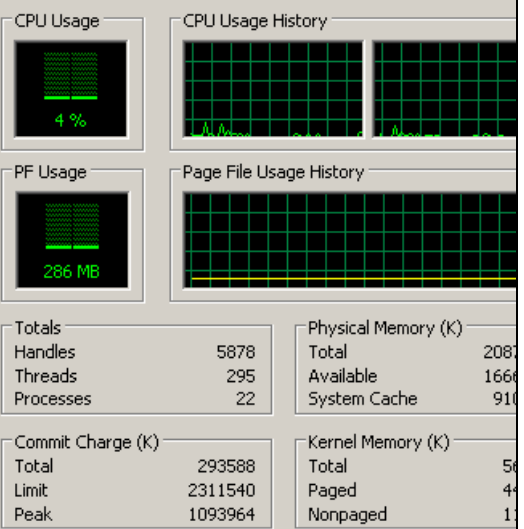
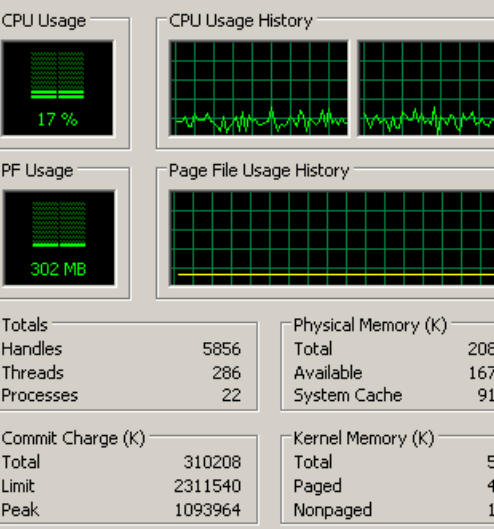
Tabel 5.7. Pengujian CPU Usage pada 1 Server (sambungan)

	
Sebelum Multicast	Saat Multicast
	
	
Sebelum Multicast	Saat Multicast

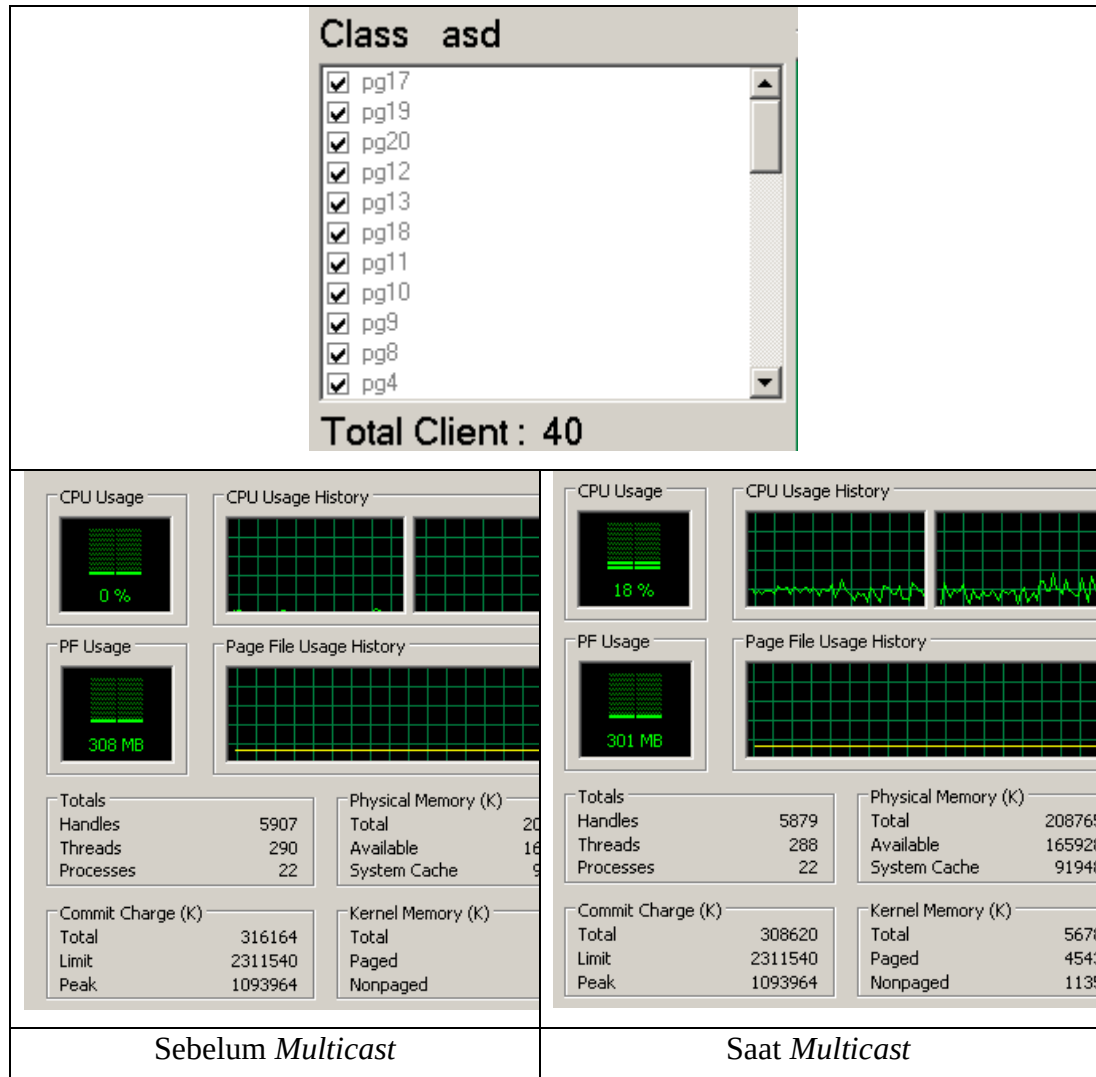
Tabel 5.7. Pengujian CPU Usage pada 1 Server (sambungan)



Tabel 5.7. Pengujian CPU Usage pada 1 Server (sambungan)

	
Sebelum Multicast	Saat Multicast
	
	
Sebelum Multicast	Saat Multicast

Tabel 5.7. Pengujian CPU Usage pada 1 Server (sambungan)



Dari Tabel 5.7, dapat disimpulkan bahwa penambahan jumlah *client* pada *server* tidak mempengaruhi CPU usage yang digunakan pada saat sebelum *multicast* maupun saat terjadi *multicast*. Pada saat sebelum *multicast*, penggunaan CPU usage normal yaitu sekitar 0% sampai dengan 4% saja. Ketika terjadi *multicast*, CPU usage meningkat karena *server* mengirimkan data berupa gambar terus menerus dan pemakainannya sekitar 17% sampai dengan 22%.

5.2.5. Pengujian Client dan Server

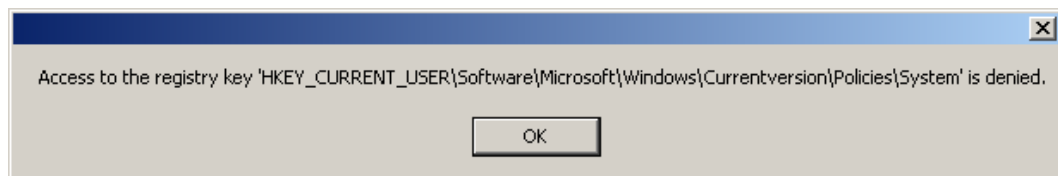
Pengujian ini merupakan pengujian antara *client* dan *server* dengan menggunakan bermacam-macam kondisi. Hal yang diujikan antara lain *password*, *priviledge*, dan penggunaan *network* yang berbeda.

5.2.5.1. Sebagai Administrator dengan Password Berbeda

Pengujian dilakukan dengan *server* yang *login* sebagai administrator dan *client* yang *login* sebagai administrator tetapi dengan *password* yang berbeda dengan *server*. Dari hasil pengujian, *client* tetap dapat *join* ke dalam *class* yang dibuat oleh *server* tanpa ada masalah.

5.2.5.2. Dengan Priviledge yang Berbeda

Server *login* sebagai administrator dan *client* *login* sebagai *user* biasa (bukan administrator). Fitur yang tidak dapat dijalankan adalah *block hot key keyboard* karena fitur ini mengubah *registry* dan Windows tidak mengizinkan kecuali pada administrator. Untuk lebih jelas dapat dilihat pada Gambar 5.22.



Gambar 5.22. Access Denied

5.2.5.3. Dengan Network yang Berbeda

Pengujian dilakukan dengan 2 *network* berbeda yaitu *server* dijalankan pada *network* kelas A dan *client* dijalankan pada *network* kelas C dan sebaliknya. Hasil yang didapatkan adalah *server* dan *client* tidak dapat tersambung karena sudah melewati sebuah *router*. Dibutuhkan sebuah *setting* khusus agar paket data yang dikirim dapat melewati *router* dengan menggunakan *multicast routing*.

5.2.6. Pengujian Remote terhadap server dan client

Pengujian lain yang dilakukan adalah melakukan Remote Desktop milik Windows terhadap *server* dan *client*. Untuk remote terhadap *server*, kendali *server* berpindah pada komputer yang melakukan remote. Untuk remote terhadap *client*, semua fungsi pada komputer yang melakukan remote tergantung pada fitur yang sedang diaktifkan oleh *server*. Misalnya, pada saat *server* mengaktifkan fitur *block keyboard* ataupun *block mouse*, maka komputer yang melakukan *remote* tetap dapat melakukan fungsi *keyboard* atau *mouse*, tetapi pada layar remote tetap

tidak dapat menjalankan fungsi *mouse* ataupun *keyboard*. Remote Desktop tidak mempengaruhi fitur yang dijalankan oleh *server* maupun *client*.

5.2.7. Pengujian dengan Program Lain

Program lain yang akan dibandingkan adalah Faronics Insight. Insight adalah sebuah teknologi yang baik untuk manajemen sebuah kelas yang dapat membantu pengajar dalam hal mengajar, mendampingi, memonitor dan berkomunikasi dengan seluruh peserta kelas dari satu komputer sentral. Pengajar tidak perlu lagi berhadapan dengan hal-hal yang dapat mengganggu kelas seperti permainan, aplikasi lain dan *web browsing* ketika sedang terjadi proses belajar mengajar. Hal-hal yang akan diujikan antara lain *delay* ketika sedang memperlihatkan tampilan pengajar, memonitor seorang pelajar dan fitur lain seperti *block application* dan *block USB*.

5.2.7.1. Pengujian Delay saat Broadcast

Insight melakukan hal yang hampir sama yaitu mengganti tampilan layar monitor *client* dengan tampilan monitor sendiri. Pada saat show, program Insight pada *client* mengganti gambar belakang dengan warna biru polos. Untuk lebih jelas, dapat dilihat pada Gambar 5.23.



Gambar 5.23. Tampilan saat Insight Show

Pengujian *delay* dilakukan sebanyak 10 kali dengan menulis pada *notepad* dan melihat seberapa lama sampai pada *client* dengan menggunakan *stopwatch*. Berikut adalah tabel data *delay* antara program penulis dengan Insight. Data ada dalam satuan waktu detik.

Tabel 5.8. Pengujian *Delay* saat *Broadcast*

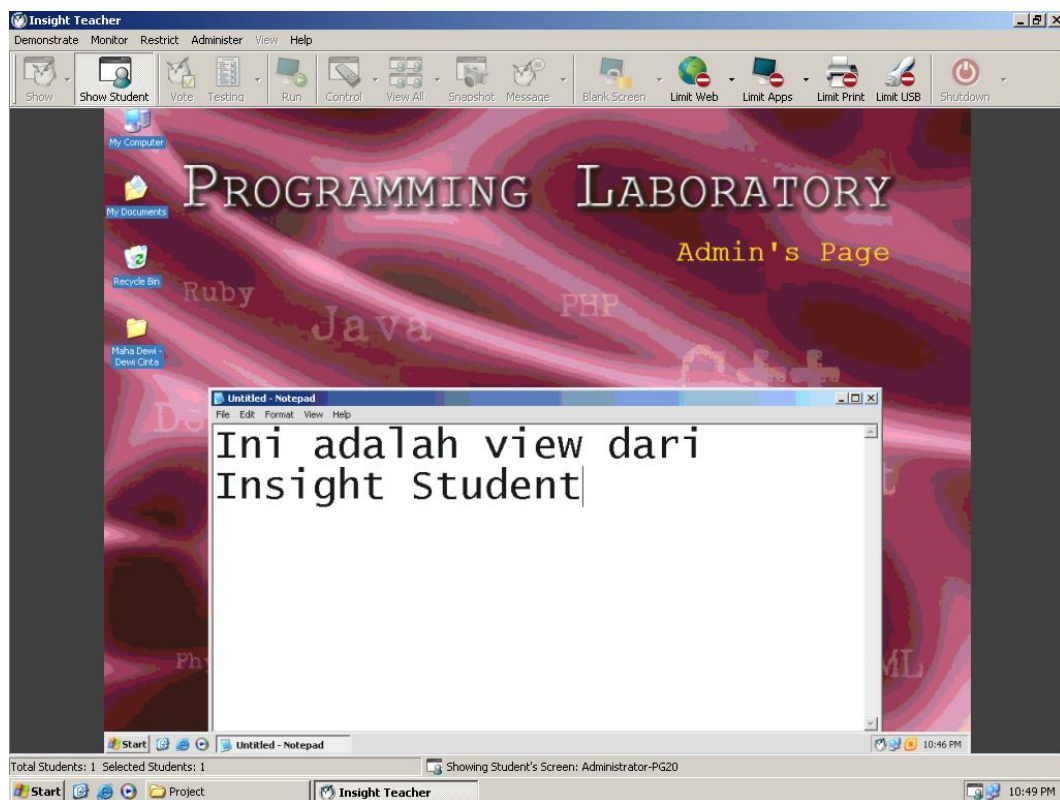
	Insight Faronic	Program Penulis
Percobaan 1	00.43	00.98
Percobaan 2	00.38	00.73
Percobaan 3	00.38	00.89
Percobaan 4	00.35	00.83
Percobaan 5	00.36	00.95
Percobaan 6	00.33	00.85
Percobaan 7	00.33	00.74
Percobaan 8	00.32	00.79
Percobaan 9	00.30	00.99

Percobaan 10	00.39	00.81
Rata – rata :	0.357	0.856

Dari data pada Tabel 5.8, dapat disimpulkan bahwa *delay* pada program Insight lebih bagus dari program yang penulis buat. Perbedaan waktu kurang lebih setengah dari program penulis.

5.2.7.2. Pengujian *Delay* saat *View Client*

Insight melakukan hal yang hampir sama yaitu mengambil tampilan *client* yang dipilih. Gambar yang diambil merupakan hasil format sehingga ada penurunan kualitas warna dan berakibat tidak sebagus tampilan *client* yang sebenarnya. Untuk lebih jelas dapat dilihat pada Gambar 5.24.



Gambar 5.24. Tampilan *View Client* Insight

Pengujian *delay* dilakukan sebanyak 10 kali percobaan dengan menulis pada *notepad* dan menghitung berapa lama waktu yang dibutuhkan untuk menampilkan pada *server* dengan menggunakan *stopwatch*. Berikut adalah tabel data *delay* saat *View Client* dalam satuan detik.

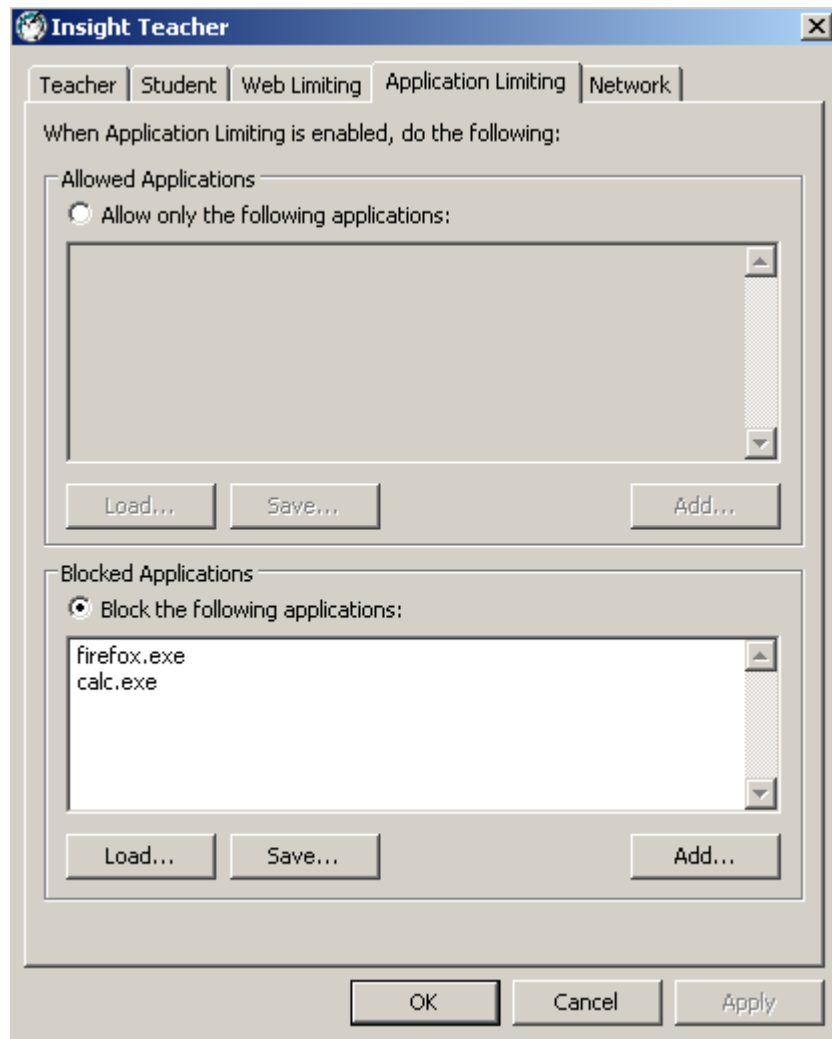
Tabel 5.9. Pengujian *Delay* saat View Client

	Insight Faronic	Program Penulis
Percobaan 1	01.16	00.62
Percobaan 2	00.84	00.80
Percobaan 3	00.84	00.80
Percobaan 4	00.72	00.79
Percobaan 5	00.99	00.74
Percobaan 6	00.63	00.62
Percobaan 7	00.89	00.69
Percobaan 8	00.80	00.63
Percobaan 9	00.76	00.62
Percobaan 10	01.03	00.73
Rata – rata :	0.866	0.704

Dari data Tabel 5.9, dapat disimpulkan bahwa program penulis lebih cepat meskipun tidak berbeda jauh dan memiliki kualitas gambar yang lebih baik dari Insight.

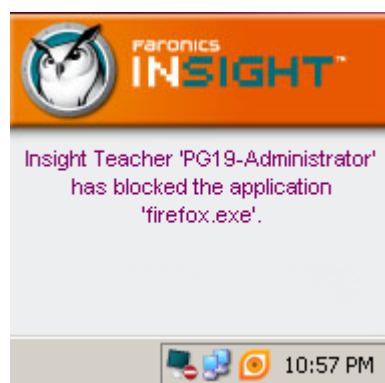
5.2.7.3. Pengujian Fitur *Block Application*

Insight juga memiliki fitur untuk *block application*. Insight memiliki *setting* yang lebih bagus. Insight meminta inputan berupa *file .exe*. Untuk lebih jelasnya, dapat dilihat pada Gambar 5.25.



Gambar 5.25. Tampilan *Setting Block Application*

Apabila tombol OK di tekan setelah memasukkan *input* berupa *file .exe*, maka pada *client* akan ditampilkan sebuah pesan yang menunjukkan bahwa *server* telah membatasi sebuah aplikasi. Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini.



Gambar 5.26. Pesan saat Block Application

Dari gambar dapat disimpulkan bahwa untuk fitur *block application*, Insight jauh lebih baik karena meminta *input* berupa *file .exe* daripada program penulis yang meminta *input* berupa nama.

5.2.7.4. Pengujian Fitur *Block USB*

Insight juga dapat melakukan Block USB. Hampir sama dengan *block application*, apabila fitur Block USB diaktifkan maka akan muncul pesan pemberitahuan pada *client* bahwa USB sedang dibatasi untuk digunakan. Untuk lebih jelasnya dapat dilihat pada Gambar 5.27.



Gambar 5.27. Pesan saat Block USB

Fitur Block USB dari Insight lebih baik dari program penulis karena pada saat ada USB yang terdeteksi maka akan langsung di *remove*. Pada program penulis, Block USB hanya membatasi pengenalan USB pada saat fitur diaktifkan dan tidak dapat me-*remove* USB saat fitur diaktifkan.

Tabel 5.1. Pengujian Seluruh Fitur pada Beberapa <i>Client</i>	72
Tabel 5.2. Pengujian <i>Delay</i> Berdasarkan Tipe Format Gambar	73
Tabel 5.3. Pengujian <i>Delay</i> Berdasarkan Resolusi Monitor	74
Tabel 5.4. Penggunaan <i>Bandwidth</i> dengan Jumlah <i>Client</i> Bervariasi	75
Tabel 5.5. Pengujian Konsumsi <i>Bandwidth</i> Berdasarkan Tipe Format Gambar ..	81
Tabel 5.6. Perbandingan Penggunaan <i>Bandwidth</i> Terhadap Tipe Format Gambar	84
Tabel 5.7. Pengujian CPU <i>Usage</i> pada 1 <i>Server</i>	86
Tabel 5.8. Pengujian <i>Delay</i> saat <i>Broadcast</i>	93
Tabel 5.9. Pengujian <i>Delay</i> saat View Client	95