

BAB IV STUDI KASUS

4.1 *Use case SIMBION*

Pada bab ini akan dibahas eksperimen utama yang dilakukan penulis. Eksperimen utama berupa pengujian terhadap studi kasus, yaitu Sistem Informasi Manajemen Beasiswa *Online* (SIMBION).

Pengguna SIMBION meliputi beberapa pihak sebagai berikut [DGMRW07]:

- **Mahasiswa**

Mahasiswa merupakan pihak yang secara langsung berkepentingan dengan sistem ini dalam hal pengajuan aplikasi dan penerimaan beasiswa.

- **Divisi Mahalum Fakultas**

Pihak ini berperan dalam pengelolaan beasiswa di tingkat fakultas. Pengelolaan beasiswa meliputi proses pengumuman penawaran beasiswa, penerimaan pengajuan beasiswa, persetujuan pengajuan, manajemen dokumen penyerta aplikasi, dan pelaporan ke pihak-pihak berkepentingan seperti Dekanat, Direktorat, dan Sponsor.

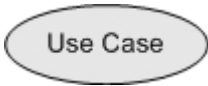

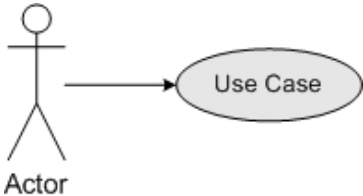
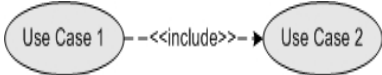

- **Subdirektorat Kesejahteraan Mahasiswa Direktorat Kemahasiswaan UI**

Pihak ini mengelola dan mengorganisasi beasiswa pada skala universitas. Secara umum, pengelolaan di tingkat ini adalah suatu bentuk tindak lanjut dari pengelolaan beasiswa pada tingkat fakultas. Oleh karena itu, secara langsung pihak ini dibantu oleh semua Divisi Mahalum untuk mendapatkan keseluruhan data dari setiap fakultas.

Untuk menjelaskan kebutuhan fungsional, tim pengembang menggunakan *use-case diagram*. *Use-case diagram* adalah sebuah diagram yang menggambarkan interaksi dan keterhubungan antara sistem yang akan dibuat dengan sistem-sistem eksternal lainnya, termasuk *user* yang akan menggunakan

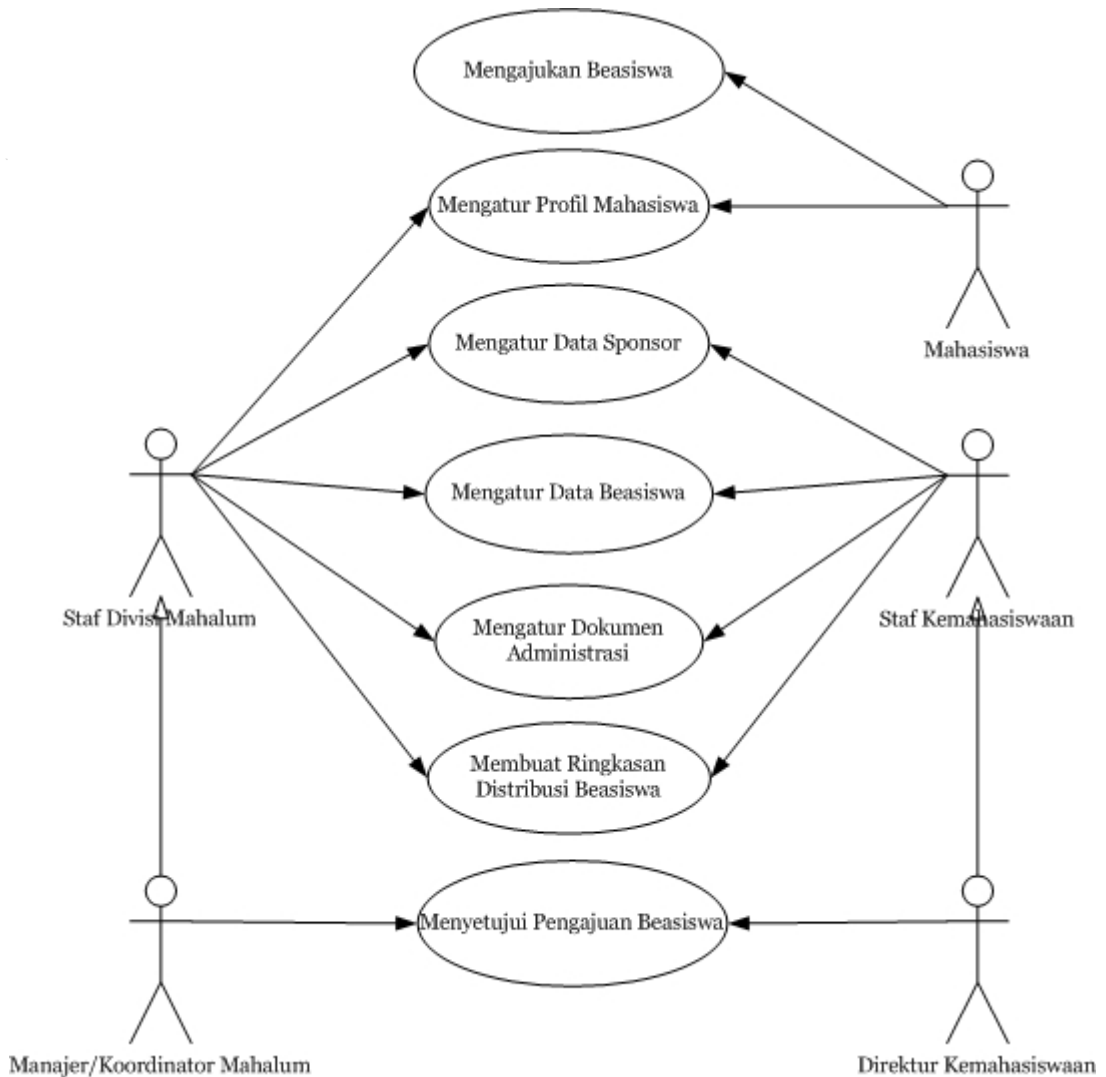
sistem tersebut. Tabel 4-1 berikut berisi notasi *use-case diagram* beserta dengan penjelasan singkat dari notasi-notasi yang digunakan.

Tabel 4-1. Use case SIMBION

Notasi	Nama	Deskripsi
	<i>Use-Case</i>	Lambang dari suatu fungsi pada sistem. <i>Use-case</i> terdiri dari skenario yang dapat dilakukan oleh pengguna terhadap sistem untuk mencapai tujuan dari suatu proses bisnis.
	<i>Actor</i>	Pengguna yang menginisiasi terjadinya skenario pada <i>use-case</i> untuk menjalankan suatu proses bisnis pada sistem.
	<i>Association</i>	Notasi yang melambangkan suatu bentuk interaksi dan relasi antara <i>Actor</i> dan <i>Use Case</i> , dimana <i>Actor</i> menggunakan atau memicu berjalannya suatu fungsi di dalam <i>use-case</i> tersebut.
	<< <i>include</i> >>	<i>Use Case 1</i> membutuhkan fungsionalitas dari <i>Use Case 2</i> .
	<i>Inheritance</i>	<i>Actor 2</i> “is a” <i>Actor 1</i> . Dengan kata lain, <i>Actor 2</i> juga memiliki karakteristik yang dipunyai oleh <i>Actor 1</i> .

Kebutuhan fungsional dari SIMBION dapat digambarkan oleh *use-case diagram*

pada Gambar.



Gambar 4-1. Use case SIMBION

Berikut ini adalah rincian *use-cases* dari SIMBION:

1. Mengatur Profil Mahasiswa

Actor dalam *use-case* ini adalah Manajer/Koordinator Mahalum, Staf Divisi Mahalum dan Mahasiswa. Pada *use-case* ini *actor* dapat melakukan pengaturan data mahasiswa yang meliputi penambahan, perubahan data profil mahasiswa. Selain itu, *actor* Manajer/Koordinator Mahalum dan Staf Divisi Mahalum juga dapat melihat, menambah, dan menghapus daftar mahasiswa.

2. Mengajukan Beasiswa

Actor dalam *use-case* ini adalah mahasiswa. *Use-case* ini menangani pengajuan beasiswa dari mahasiswa dengan memilih beasiswa yang masih dapat dipilih (dalam masa aktif pengajuan). Setiap mahasiswa dapat memilih lebih dari satu beasiswa.

3. Menyetujui Pengajuan Beasiswa

Actor dalam *use-case* ini adalah Manajer/Koordinator Mahalum maupun Direktur Kemahasiswaan. Pada *use-case* ini Manajer/Koordinator Mahalum mencocokkan data mahasiswa dengan persyaratan beasiswa. Mahasiswa yang paling memenuhi persyaratan disetujui oleh Manajer/Koordinator Mahalum untuk diteruskan ke tahap pengelolaan beasiswa selanjutnya, baik yang dilakukan oleh pihak Direktorat Kemahasiswaan maupun pihak sponsor.

4. Mengatur Data Sponsor

Actor dalam *use-case* ini adalah Manajer/Koordinator dan Staf Divisi Mahalum beserta pihak Direktorat Kemahasiswaan. Pada *use-case* ini *actor* dapat melakukan pengaturan data sponsor yang meliputi penambahan, pengubahan, dan penghapusan detil data sponsor.

5. Mengatur Data Beasiswa

Actor dalam *use-case* ini adalah Manajer/Koordinator dan Staf Divisi Mahalum beserta pihak Direktorat Kemahasiswaan. Pada *use-case* ini *actor* dapat melakukan pengaturan data beasiswa yang meliputi penambahan, pengubahan, dan penghapusan data beasiswa. Selain itu, *actor* juga dapat mengatur penambahan, pengubahan dan penghapusan periode beasiswa.

6. Mengatur Dokumen

Actor dalam *use-case* ini adalah Manajer/Koordinator dan Staf Divisi Mahalum beserta pihak Direktorat Kemahasiswaan. Pada *use-case* ini *actor* dapat melakukan pengaturan dokumen yang meliputi penambahan, pengubahan, dan penghapusan dokumen. Dokumen terdiri dari surat pengantar laporan, surat pengantar dan surat rekomendasi. Untuk surat

pengantar dan surat rekomendasi disertakan bersama dokumen-dokumen mahasiswa. Sedangkan untuk surat pengantar laporan akan disertakan bersama ringkasan beasiswa yang telah dicetak.

7. Membuat Ringkasan Distribusi Beasiswa

Actor dalam *use-case* ini adalah Manajer/Koordinator dan Staf Divisi Mahalum beserta pihak Direktorat Kemahasiswaan. *Use-case* ini memungkinkan *actor* untuk melihat dan mencetak ke dalam format .PDF ringkasan beasiswa-beasiswa yang tersebar di domain mereka, baik ditingkat fakultas maupun universitas. Ringkasan ini dibuat berdasarkan kategorisasi yang telah dibuat dalam sistem ini.

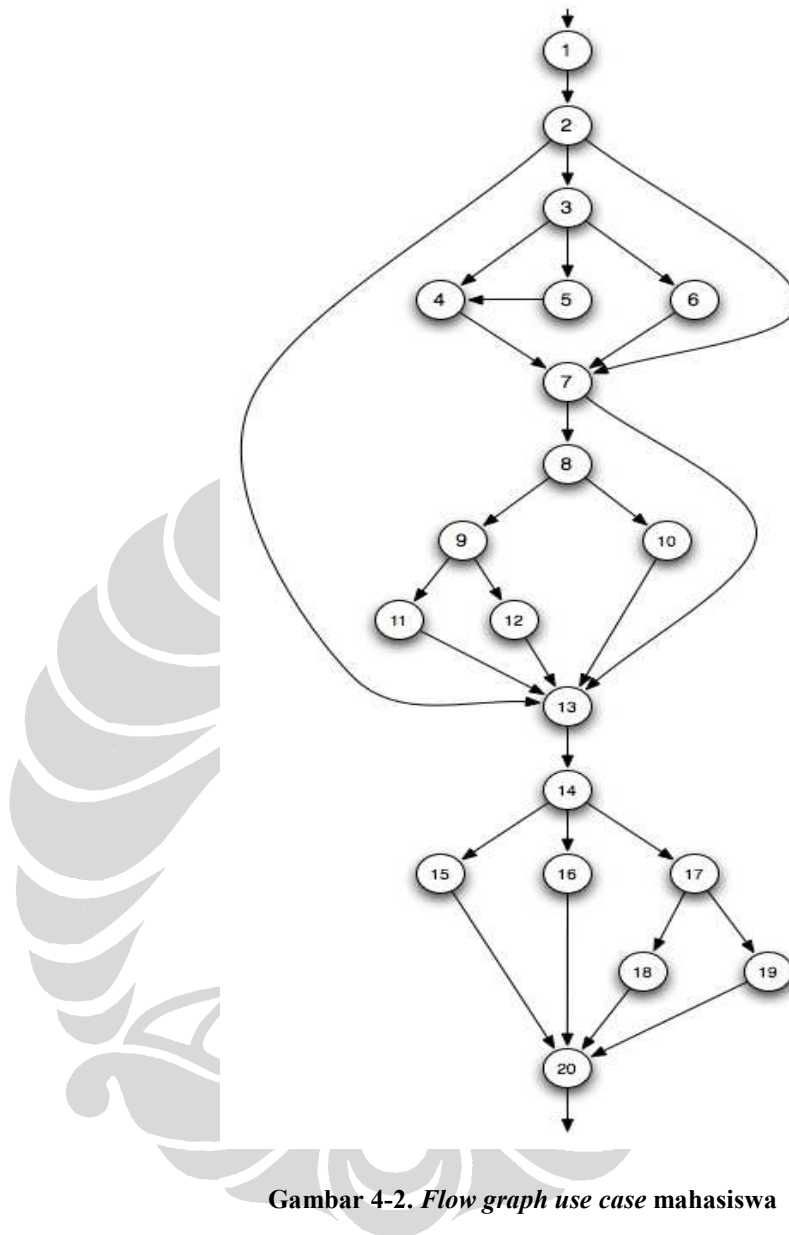
Pada pengujian yang dilakukan, akan digunakan *use case* dari peran mahasiswa. Menurut *use case* yang ada, fungsi-fungsi yang bisa dilakukan oleh mahasiswa adalah:

- Log in.
- Melihat profil.
- Melihat daftar beasiswa.
- Mengajukan beasiswa.
- Mengubah pengajuan beasiswa.
- Melihat detil daftar beasiswa.
- Membatalkan pengajuan beasiswa.
- Melihat daftar pengajuan beasiswa.
- Melihat detil daftar pengajuan beasiswa.
- Log out.

4.2 Penyusunan *test case*

Dari *use case* yang ada akan dibuat sebuah *graph* yang menggambarkan aktivitas pada aktor mahasiswa. *Graph* akan dibuat dengan asumsi seluruh proses hanya berjalan searah, tidak ada proses yang *looping*, dan minimal setiap *node* dilewati sekali. Tujuan pembuatan *graph* adalah untuk menentukan minimum *path coverage* untuk mencapai *full line coverage*. Dengan begitu *test case* yang akan dibuat adalah kumpulan *test case* minimum yang dapat memenuhi seluruh

fungsi yang ada. Bentuk *graph* dari fungsi mahasiswa adalah:



Gambar 4-2. *Flow graph use case mahasiswa*

Keterangan *node* dari *graph*:

1. Log in
2. Melihat profil mahasiswa
3. Ubah profil mahasiswa
4. Simpan
5. Load dari SIAK-NG
6. Batal
7. Daftar beasiswa

8. Lihat detil beasiswa
9. Mengajukan beasiswa
10. Kembali
11. Simpan
12. Batal
13. Daftar pengajuan beasiswa
14. Melihat detil data pengajuan beasiswa
15. Kembali
16. Membatalkan pengajuan
17. Ubah pengajuan
18. Simpan
19. Batal
20. Log out

Berdasarkan *flow graph* yang ada di atas, terdapat 68 kemungkinan *path* yang ada dengan jumlah *edge* sebanyak 29. Kemungkinan seluruh *path* yang ada adalah:

Tabel 4-2. Daftar *path*

No.	<i>Path</i>
1.	1-2-3-4-7-8-9-11-13-14-15-20
2.	1-2-3-5-4-7-8-9-11-13-14-15-20
3.	1-2-3-6-7-8-9-11-13-14-15-20
4.	1-2-3-4-7-8-9-12-13-14-15-20
5.	1-2-3-5-4-7-8-9-12-13-14-15-20
6.	1-2-3-6-7-8-9-12-13-14-15-20
7.	1-2-3-4-7-8-10-13-14-15-20
8.	1-2-3-5-4-7-8-10-13-14-15-20
9.	1-2-3-6-7-8-10-13-14-15-20
10.	1-2-3-4-7-8-9-11-13-14-16-20
11.	1-2-3-5-4-7-8-9-11-13-14-16-20
12.	1-2-3-6-7-8-9-11-13-14-16-20
13.	1-2-3-4-7-8-9-12-13-14-16-20

14.	1-2-3-5-4-7-8-9-12-13-14-16-20
15.	1-2-3-6-7-8-9-12-13-14-16-20
16.	1-2-3-4-7-8-10-13-14-16-20
17.	1-2-3-5-4-7-8-10-13-14-16-20
18.	1-2-3-6-7-8-10-13-14-16-20
19.	1-2-3-4-7-8-9-11-13-14-17-18-20
20.	1-2-3-5-4-7-8-9-11-13-14-17-18-20
21.	1-2-3-6-7-8-9-11-13-14-17-18-20
22.	1-2-3-4-7-8-9-12-13-14-17-18-20
23.	1-2-3-5-4-7-8-9-12-13-14-17-18-20
24.	1-2-3-6-7-8-9-12-13-14-17-18-20
25.	1-2-3-4-7-8-10-13-14-17-18-20
26.	1-2-3-5-4-7-8-10-13-14-17-18-20
27.	1-2-3-6-7-8-10-13-14-17-18-20
28.	1-2-3-4-7-8-9-11-13-14-17-19-20
29.	1-2-3-5-4-7-8-9-11-13-14-17-19-20
30.	1-2-3-6-7-8-9-11-13-14-17-19-20
31.	1-2-3-4-7-8-9-12-13-14-17-19-20
32.	1-2-3-5-4-7-8-9-12-13-14-17-19-20
33.	1-2-3-6-7-8-9-12-13-14-17-19-20
34.	1-2-3-4-7-8-10-13-14-17-19-20
35.	1-2-3-5-4-7-8-10-13-14-17-19-20
36.	1-2-3-6-7-8-10-13-14-17-19-20
37.	1-2-13-14-15-20
38.	1-2-13-14-16-20
39.	1-2-13-14-17-18-20
40.	1-2-13-14-17-19-20
41.	1-2-7-13-14-15-20
42.	1-2-7-13-14-16-20
43.	1-2-7-13-14-17-18-20

44.	1-2-7-13-14-17-19-20
45.	1-2-7-8-9-11-13-14-15-20
46.	1-2-7-8-9-12-13-14-16-20
47.	1-2-7-8-10-13-14-17-18-20
48.	1-2-7-8-9-11-13-14-17-19-20
49.	1-2-7-8-9-11-13-14-15-20
50.	1-2-7-8-9-12-13-14-16-20
51.	1-2-7-8-10-13-14-17-18-20
52.	1-2-7-8-9-11-13-14-17-19-20
53.	1-2-7-8-9-11-13-14-15-20
54.	1-2-7-8-9-12-13-14-16-20
55.	1-2-7-8-10-13-14-17-18-20
56.	1-2-7-8-9-11-13-14-17-19-20
57.	1-2-3-4-7-13-14-15-20
58.	1-2-3-4-7-13-14-16-20
59.	1-2-3-4-7-13-14-17-18-20
60.	1-2-3-4-7-13-14-17-19-20
61.	1-2-3-5-4-7-13-14-15-20
62.	1-2-3-5-4-7-13-14-16-20
63.	1-2-3-5-4-7-13-14-17-18-20
64.	1-2-3-5-4-7-13-14-17-19-20
65.	1-2-3-6-13-14-15-20
66.	1-2-3-6-13-14-16-20
67.	1-2-3-6-13-14-17-18-20
68.	1-2-3-6-13-14-17-19-20

Dari daftar *path* yang ada diatas, dapat dibuat daftar kumpulan *path* minimum. Kumpulan *path* minimum untuk mencapai kondisi *full line coverage* adalah:

Tabel 4-3. Minimum path mahasiswa

No.	<i>Path</i>
-----	-------------

20.	1-2-3-5-4-7-8-9-11-13-14-17-18-20
33.	1-2-3-6-7-8-9-12-13-14-17-19-20
16.	1-2-3-4-7-8-10-13-14-16-20
41.	1-2-7-13-14-15-20

Dengan menggunakan rumus *cyclomatic complexity metric*, dapat dihitung nilai kompleksitas dari program yang akan diuji [GAL04]. Perhitungannya adalah sebagai berikut:

$$E = 29$$

$$N = 20$$

$$V(G) = E - N + 2$$

$$V(G) = 29 - 20 + 2 = 11$$

Dari hasil perhitungan, nilai kompleksitas untuk *use case* mahasiswa adalah 11. Nilai tersebut termasuk kategori tidak terlalu sulit [GAL04]. Nilai kompleksitas yang ada adalah jumlah maksimum *path* yang dapat dibentuk. Jumlah set maksimum dari *path* yang ada:

No.	<i>Path</i>	<i>Edge</i> yang ditambahkan	Jumlah <i>Edge</i> yang ditambahkan
1.	1-2-3-4-7-8-9-11-13-14-15-20	1-2, 2-3, 3-4, 8-9, 9-11, 11-13, 13-14, 14-15, 15-20	9
2.	1-2-3-4-7-8-9-11-13-14-16-20	4-7, 7-8, 14-16, 16-20	4
3.	1-2-3-4-7-8-9-11-13-14-17-18-20	14-17, 17-18, 18-20	3
4.	1-2-3-4-7-8-9-11-13-14-17-19-20	17-19, 19-20	2
5.	1-2-3-5-4-7-8-	5-4	1

	9-11-13-14-15-20		
6.	1-2-3-6-7-8-9-11-13-14-15-20	6-7	1
7.	1-2-3-6-7-8-9-12-13-14-16-20	9-12, 12-13	2
8.	1-2-3-6-10-13-14-15-16-20	3-6, 6-10, 10-13, 15-16	4
9.	1-2-13-14-15-20	2-13	1
10.	1-2-7-8-10-13-14-16-20	2-7	1
11.	1-2-7-13-14-15-20	7-13	1

Tabel 4-4. Maksimum path

Berdasarkan minimum *path* yang didapat, maka kumpulan *test case* yang akan diuji adalah sebagai berikut:

- Log in-profil-ubah profil-load dari SIAK-NG-simpan-daftar beasiswa-detil beasiswa-ajukan beasiswa-simpan-daftar pengajuan-detil daftar pengajuan-ubah detil daftar pengajuan-simpan-log out.
- Log in-profil-ubah profil-batal-daftar beasiswa-detil beasiswa-ajukan beasiswa-batal-daftar pengajuan-detil daftar pengajuan-ubah detil daftar pengajuan-batal-log out.
- Log in-profil-ubah profil-simpan-daftar beasiswa-detil beasiswa-kembali-daftar pengajuan-detil daftar pengajuan-batalkan beasiswa-log out.
- Log in-profil-daftar beasiswa-daftar pengajuan-detil daftar pengajuan-kembali-log out.

Kumpulan *test case* tersebut akan digunakan untuk melakukan pengujian dengan menggunakan Rational Test Manager.

BAB V HASIL UJI COBA

Pada bab ini akan dipaparkan proses pengujian yang dilakukan pada SIMBION. Hasil pengujian dan interpretasinya akan dijabarkan lebih lanjut. Pengujian akan berfokus pada kumpulan *test case* yang telah dihasilkan sebelumnya.

5.1 Perencanaan pengujian

Pengujian akan menggunakan Rational TestManager dan Robot. Untuk itu langkah pertama yang harus dilakukan adalah meregistrasi proyek pengujian pada Rational Administrator. Setelah terdaftar pada Rational Administrator, fungsi-fungsi *software* Rational lainnya akan bisa digunakan. Proses registrasi proyek pengujian menggunakan nama Simbion Final. Proyek ini akan menggunakan database Microsoft Access untuk menyimpan data selama proses pengujian.

Langkah berikutnya adalah membuat *test plan* dari kumpulan *test case* yang ada sebelumnya. Test plan yang dibuat berdasarkan pada *use case* mahasiswa. Fungsi-fungsi yang akan dimasukkan dalam *test plan* adalah fungsi-fungsi yang memenuhi minimal *full line coverage*. ada 4 kumpulan fungsi yang akan digunakan, yaitu:

- Log in>profil>ubah profil>load dari SIAK-NG>simpan profil>daftar beasiswa>detil beasiswa>ajukan beasiswa>simpan beasiswa yang diajukan>daftar pengajuan beasiswa>detil daftar pengajuan beasiswa>ubah detil daftar pengajuan>simpan-log out.
- Log in>profil>ubah profil>membatalkan ubah profil>daftar beasiswa>detil beasiswa>ajukan beasiswa>membatalkan mengajukan beasiswa>daftar pengajuan beasiswa>detil daftar pengajuan beasiswa>ubah detil daftar pengajuan>membatalkan perubahan detil pengajuan>log out.
- Log in>profil>ubah profil>simpan profil>daftar beasiswa>detil beasiswa>kembali ke daftar beasiswa>daftar pengajuan beasiswa>detil daftar pengajuan beasiswa>membatalkan beasiswa>log out.
- Log in>profil>daftar beasiswa>daftar pengajuan beasiswa>detil daftar pengajuan beasiswa>kembali ke daftar pengajuan beasiswa>log out.

5.2 Perekaman Robot

Setelah seluruh test plan dibuat, langkah berikutnya adalah merekam *test script* menggunakan Rational Robot. *Test script* dibuat berdasarkan kumpulan fungsi yang ada. Nantinya, *test script* akan digunakan untuk melakukan pengujian. Langkah untuk merekam menggunakan Robot telah dijelaskan dalam bab sebelumnya. Berikut adalah daftar *test script* yang dibuat:

- Log in
- Melihat profil
- Mengubah profil
- Load profil dari SIAK-NG
- Menyimpan profil
- Membatalkan perubahan profil.
- Daftar beasiswa
- Detil beasiswa
- Mengajukan beasiswa
- Kembali ke daftar beasiswa
- Menyimpan pengajuan beasiswa
- Membatalkan pengajuan beasiswa
- Daftar pengajuan beasiswa
- Detil beasiswa yang diajukan
- Kembali ke daftar pengajuan beasiswa
- Membatalkan beasiswa yang telah diajukan
- Mengubah detil beasiswa yang diajukan
- Menyimpan perubahan detil beasiswa yang diajukan
- Membatalkan perubahan detil beasiswa yang diajukan
- Log out

5.3 *Functional testing* pada *Local computer*

Setelah semua *test script* yang diperlukan direkam, dapat dilakukan proses *functional testing*. Pelaksanaan *functional testing* dapat dilakukan dengan 2 cara, yaitu melakukan pengujian menggunakan *Local computer* dan menggunakan

virtual tester. Pada bagian ini, akan dijelaskan bagaimana melakukan *functional testing* pada *Local computer*. Untuk melakukan pengujian tersebut, sebelumnya harus dibuat *test suite* yang nantinya akan dijalankan untuk melakukan pengujian. *Test suite* untuk *functional testing* dapat dibuat berdasarkan *test plan* atau langsung berdasarkan *test script*. Jumlah *test suite* yang dibuat berdasarkan jumlah minimum *path* yang telah dibuat sebelumnya. Ada 4 *test suite* yang akan dibuat, yaitu:

- Uji coba fungsi SIMBION final. *Test suite* ini akan berisi *test script*:
 - Log in
 - Profil
 - Mengubah profil
 - Load dari SIAK-NG
 - Menyimpan profil
 - Daftar beasiswa
 - Detil beasiswa
 - Mengajukan beasiswa
 - Menyimpan pengajuan beasiswa
 - Daftar pengajuan beasiswa
 - Detil daftar pengajuan beasiswa
 - Mengubah detil beasiswa yang diajukan
 - Menyimpan perubahan detil beasiswa yang diajukan
 - Log out
- Uji coba fungsi SIMBION final 2. *Test suite* ini akan berisi *test script*:
 - Log in
 - Profil
 - Mengubah profil
 - Membatalkan perubahan profil
 - Daftar beasiswa
 - Detil beasiswa
 - Mengajukan beasiswa
 - Membatalkan pengajuan beasiswa
 - Daftar pengajuan beasiswa

- Detil daftar pengajuan beasiswa
 - Mengubah detil beasiswa yang diajukan
 - Membatalkan perubahan detil beasiswa yang diajukan
 - Log out.
- Uji coba fungsi SIMBION final 3. *Test suite* ini akan berisi test sript:
 - Log in
 - Profil
 - Mengubah profil
 - Menyimpan profil
 - Daftar beasiswa
 - Detil beasiswa
 - Kembali ke daftar beasiswa
 - Daftar pengajuan beasiswa
 - Detil daftar pengajuan beasiswa
 - Membatalkan beasiswa yang telah dilakukan
 - Log out.
 - Uji coba fungsi SIMBION final 4. *Test suite* ini akan berisi *test script*:
 - Log in
 - Profil
 - Daftar beasiswa
 - Daftar pengajuan beasiswa
 - Detil daftar pengajuan beasiswa
 - Kembali ke daftar pengajuan beasiswa
 - Log out.

Pengujian dilakukan dengan menjalankan masing-masing *test suite* yang ada. Hasil dari pengujian menunjukkan bahwa setiap fungsi berjalan dengan baik. Hal tersebut diindikasikan dengan status *Pass* yang ada pada *test log report*. Berikut adalah laporan hasil pengujian dari masing-masing *test suite*:

- Uji coba fungsi SIMBION final
 - Test suite ini berhasil dieksekusi sepenuhnya. Seluruh *test script* yang dijalankan berstatus *Pass*. Status *warning* yang ada bukan berarti

pengujian gagal, tetapi berarti ada aplikasi lain yang berjalan pada saat pengujian. Hasil pengujian dapat dilihat pada gambar 5-1.

Event Type	Result	Date & Time	Failure
Suite Start (Uji coba fungsi SIMBION Final)	Pass	12/21/2008 1:30:12 ...	
Computer Start (Suite 1 [1])	Warning	12/21/2008 1:30:14 ...	
Script Start (Log in)	Pass	12/21/2008 1:30:17 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:22 ...	
Script End (Log in)	Pass	12/21/2008 1:30:22 ...	
Script Start (Profil)	Pass	12/21/2008 1:30:22 ...	
Playback Warning	Warning	12/21/2008 1:30:53 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:53 ...	
Script End (Profil)	Pass	12/21/2008 1:30:54 ...	
Script Start (Ubah profil)	Pass	12/21/2008 1:30:54 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:55 ...	
Script End (Ubah profil)	Pass	12/21/2008 1:30:55 ...	
Script Start (Load profil dari siak)	Pass	12/21/2008 1:30:55 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:56 ...	
Script End (Load profil dari siak)	Pass	12/21/2008 1:30:57 ...	
Script Start (Simpan profil)	Pass	12/21/2008 1:30:57 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:58 ...	
Script End (Simpan profil)	Pass	12/21/2008 1:30:58 ...	
Script Start (Daftar beasiswa)	Pass	12/21/2008 1:30:58 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:30:59 ...	
Script End (Daftar beasiswa)	Pass	12/21/2008 1:30:59 ...	
Script Start (Lihat detail daftar beasiswa)	Pass	12/21/2008 1:30:59 ...	
Playback Warning	Warning	12/21/2008 1:31:30 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:31:30 ...	
Script End (Lihat detail daftar beasiswa)	Pass	12/21/2008 1:31:30 ...	
Script Start (mengajukan beasiswa)	Pass	12/21/2008 1:31:30 ...	
Script Start (Simpan pengajuan beasiswa)	Pass	12/21/2008 1:32:02 ...	
Verification Point (Object Data - Object ...)	Pass	12/21/2008 1:32:04 ...	
Script End (Simpan pengajuan beasiswa)	Pass	12/21/2008 1:32:04 ...	

Gambar 5-1. Test log uji coba fungsi SIMBION final

Pada pengujian ini, hanya satu *virtual tester* yang berjalan. Laporan *virtual tester* yang berjalan dapat dilihat pada gambar 5-2.

```
==> Start of data for Virtual Tester: Suite 1[1] <==
==> End of data for Virtual Tester: Suite 1[1] <==
```

Gambar 5-2. Command data report uji coba fungsi SIMBION final

Pada hasil pengujian dapat dilihat juga *command trace report* yang menjabarkan aktivitas selama proses pengujian. Pada laporan tersebut akan

terlihat *time stamp* dari emulasi yang dilakukan. *Command status report* dapat dilihat pada gambar 5-3.

```
Trace of Emulation Session Activity
Total Number of Virtual Testers Emulated: 1
Number of Virtual Testers in Report: 1

Information for Virtual Tester: Suite 1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 13:30:12 2008
Time of Day Reference for Timestamps: 1229841012

Virtual Tester: Suite 1[1]      Test Script: Log in
Beginning timestamp of test script Log in:      5874
Beginning timestamp of test script Profil:      10875
Beginning timestamp of test script Ubah profil:  42125
Beginning timestamp of test script Load profil dari siak: 43499
Beginning timestamp of test script Simpan profil: 45030
Beginning timestamp of test script Daftar beasiswa: 46749
Beginning timestamp of test script Lihat detail daftar beasiswa: 47686
Beginning timestamp of test script mengajukan beasiswa: 78812
Beginning timestamp of test script Simpan pengajuan beasiswa: 110436
Beginning timestamp of test script Lihat detail daftar pengajuan beasiswa: 112217
Beginning timestamp of test script Ubah detail daftar pengajuan beasiswa: 113125
Beginning timestamp of test script Simpan perubahan detail daftar pengajuan: 114375
Beginning timestamp of test script Log out:     146296
```

admin | Currently Selected Filter:

Gambar 5-3. Command trace report uji coba fungsi SIMBION final

Dari hasil pengujian, dapat juga dilihat *command usage report* pada gambar 5-4.

```
Time of Emulation Session: Sun Dec 21 2008 13:30

                          Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:                0.0 secs  Inactive Time:                0.0 secs

Commands:                   0          Average Throughput: UNDEFN cmds/min
Passed Responses:           0          Average Throughput: UNDEFN resps/min
Failed Responses:           0          Average Throughput: UNDEFN resps/min

Time Spent Waiting:         0.0 seconds => UNDEFN
Time Spent in Input:        0.0 seconds => UNDEFN
Time Spent Thinking:        0.0 seconds => UNDEFN

Summary Statistics

Duration of Run:             0.0 seconds (00:00:00.0)

Summary Statistics for Selected Virtual Testers and Scripts:

Commands:                   0          Total Throughput: UNDEFN cmds/min

Number of Virtual Testers:  1          Number of stop_time Cmds:  0
Number of Completed Scripts: 13       Number of Uncompleted Scripts: 0

Average Number of Scripts Completed per Virtual Tester: 13.0 scripts
Average Duration for Completed Scripts: 0.0 seconds
Script Throughput for Completed Scripts: UNDEFN scripts/hour
```

Gambar 5-4. Command usage report uji coba fungsi SIMBION final

- Uji coba fungsi SIMBION final 2

Pengujian berjalan sempurna. Seluruh status dari *test script* adalah *Pass*.

Test log hasil pengujian dapat dilihat pada gambar 5-5.

Event Type	Result	Date & Time	Failure R
[-] Suite Start (Uji coba fungsi SIMBION Final 2)	Pass	12/21/2008 2:05:38 ...	
[-] Computer Start (Suite 1 [1])	Warning	12/21/2008 2:05:39 ...	
[+] Script Start (Log in)	Pass	12/21/2008 2:05:41 ...	
[+] Script Start (Profil)	Pass	12/21/2008 2:05:44 ...	
[+] Script Start (Ubah profil)	Pass	12/21/2008 2:06:15 ...	
[+] Script Start (Membatalkan ubah profil)	Pass	12/21/2008 2:06:16 ...	
[+] Script Start (Daftar beasiswa)	Pass	12/21/2008 2:06:18 ...	
[+] Script Start (Lihat detil daftar beasiswa)	Pass	12/21/2008 2:06:18 ...	
[+] Script Start (mengajukan beasiswa)	Pass	12/21/2008 2:06:49 ...	
[+] Script Start (Membatalkan pengajuan dari d...)	Pass	12/21/2008 2:07:21 ...	
[+] Script Start (Lihat detil daftar pengajuan bea...)	Pass	12/21/2008 2:07:53 ...	
[+] Script Start (Ubah detil daftar pengajuan be...)	Pass	12/21/2008 2:07:53 ...	
[+] Script Start (Membatalkan ubah detil daftar ...)	Pass	12/21/2008 2:07:55 ...	
[+] Script Start (Log out)	Pass	12/21/2008 2:07:56 ...	
Computer End	Warning	12/21/2008 2:07:57 ...	
Suite End (Uji coba fungsi SIMBION Final 2)	Pass	12/21/2008 2:07:59 ...	

Gambar 5-5. Test log uji coba fungsi SIMBION final 2

Command data report yang ada juga menunjukkan bahwa *test suite* telah dijalankan dengan baik seperti pada gambar 5-6.

```
==> Start of data for Virtual Tester: Suite 1[1] <==
==> End of data for Virtual Tester: Suite 1[1] <==
```

Gambar 5-6. Command data report uji coba fungsi SIMBION final 2

Command trace report yang ada menunjukkan *time stamp* dari emulasi yang dilakukan. *Command trace report* dapat dilihat pada gambar 5-7.

```
|
Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 1
Number of Virtual Testers in Report: 1

Information for Virtual Tester: Suite 1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 14:05:38 2008
Time of Day Reference for Timestamps: 1229843138

Virtual Tester: Suite 1[1]          Test Script: Log in
Beginning timestamp of test script Log in:      3797
Beginning timestamp of test script Profil:      6296
Beginning timestamp of test script Ubah profil:  37422
Beginning timestamp of test script Membatalkan ubah profil:  38718
Beginning timestamp of test script Daftar beasiswa:  40031
Beginning timestamp of test script Lihat detil daftar beasiswa:  40967
Beginning timestamp of test script mengajukan beasiswa:  71999
Beginning timestamp of test script Membatalkan pengajuan dari daftar beasiswa:  103561
Beginning timestamp of test script Lihat detil daftar pengajuan beasiswa2:  135046
Beginning timestamp of test script Ubah detil daftar pengajuan beasiswa2:  135906
Beginning timestamp of test script Membatalkan ubah detil daftar pengajuan2:  137203
Beginning timestamp of test script Log out:      138467
```

Gambar 5-7. Command trace report uji coba fungsi SIMBION final 2

Command usage report akan menampilkan semua data dari perintah emulasi dan respon. Pada *functional testing*, *command usage report* akan melaporkan jumlah *test script* yang berhasil dieksekusi seperti pada gambar 5-8.

```

Time of Emulation Session: Sun Dec 21 2008 14:05

                          Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:                0.0 secs  Inactive Time:                0.0 secs

Commands:                   0          Average Throughput: UNDEFN cmds/min
Passed Responses:          0          Average Throughput: UNDEFN resps/min
Failed Responses:          0          Average Throughput: UNDEFN resps/min

Time Spent Waiting:         0.0 seconds => UNDEFN
Time Spent in Input:        0.0 seconds => UNDEFN
Time Spent Thinking:        0.0 seconds => UNDEFN

Summary Statistics

Duration of Run:             0.0 seconds (00:00:00.0)

Summary Statistics for Selected Virtual Testers and Scripts:

Commands:                   0          Total Throughput: UNDEFN cmds/min

Number of Virtual Testers:  1          Number of stop_time Cnds:      0
Number of Completed Scripts: 12       Number of Uncompleted Scripts: 0

Average Number of Scripts Completed per Virtual Tester: 12.0 scripts
Average Duration for Completed Scripts: 0.0 seconds
Script Throughput for Completed Scripts: UNDEFN scripts/hour

```

Gambar 5-8. Command usage report uji coba fungsi SIMBION final 2

- Uji coba fungsi SIMBION final 3

Hasil pengujian *test suite* ini sama dengan pengujian sebelumnya. Semua berstatus *Pass*, seperti pada gambar 5-9.

Event Type	Result	Date & Time	Failure R
[-] Suite Start (Uji coba fungsi SIMBION Final 3)	Pass	12/21/2008 2:25:20 ...	
[-] Computer Start (Suite 1 [1])	Warning	12/21/2008 2:25:21 ...	
[-] Script Start (Log in)	Pass	12/21/2008 2:25:23 ...	
[-] Script Start (Profil)	Pass	12/21/2008 2:25:26 ...	
[-] Script Start (Ubah profil)	Pass	12/21/2008 2:25:57 ...	
[-] Script Start (Simpan profil)	Pass	12/21/2008 2:25:58 ...	
[-] Script Start (Daftar beasiswa)	Pass	12/21/2008 2:26:00 ...	
[-] Script Start (Lihat detil daftar beasiswa)	Pass	12/21/2008 2:26:01 ...	
[-] Script Start (Kembali ke detil mahasiswa)	Pass	12/21/2008 2:26:32 ...	
[-] Script Start (Daftar Pengajuan Beasiswa)	Pass	12/21/2008 2:26:33 ...	
[-] Script Start (Lihat detil daftar pengajuan bea...	Pass	12/21/2008 2:27:04 ...	
[-] Script Start (Membatalkan pengajuan beasis...	Pass	12/21/2008 2:27:05 ...	
[-] Script Start (Log out)	Pass	12/21/2008 2:27:38 ...	
[-] Computer End	Warning	12/21/2008 2:27:38 ...	
[-] Suite End (Uji coba fungsi SIMBION Final 3)	Pass	12/21/2008 2:27:42 ...	

Gambar 5-9. Test log uji coba fungsi SIMBION final 3

Command data report menunjukkan hanya ada satu virtual tester dan pengujian berhasil, spseti pada gambar 5-10.

```
==> Start of data for Virtual Tester: Suite 1[1] <==
==> End of data for Virtual Tester: Suite 1[1] <==
```

Gambar 5-10. Command data report uji coba fungsi SIMBION final 3

Command trace report dari test suite ini dapat dilihat pada gambar 5-11.

```
Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 1
Number of Virtual Testers in Report: 1

Information for Virtual Tester: Suite 1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 14:25:20 2008
Time of Day Reference for Timestamps: 1229844320

Virtual Tester: Suite 1[1]          Test Script: Log in

Beginning timestamp of test script Log in:      3797
Beginning timestamp of test script Profil:      6220
Beginning timestamp of test script Ubah profil: 37361
Beginning timestamp of test script Simpan profil: 38689
Beginning timestamp of test script Daftar beasiswa: 40407
Beginning timestamp of test script Lihat detil daftar beasiswa: 41282
Beginning timestamp of test script Kembali ke detil mahasiswa: 72408
Beginning timestamp of test script Daftar Pengajuan Beasiswa: 73688
Beginning timestamp of test script Lihat detil daftar pengajuan beasiswa2: 104844
Beginning timestamp of test script Membatalkan pengajuan beasiswa2: 105703
Beginning timestamp of test script Log out:    138267
```

Gambar 5-11. Command trace report uji coba fungsi SIMBION final 3

Command usage report akan menampilkan semua data dari perintah emulasi dan respon. Pada pengujian ini *command usage report* melaporkan ada 12 *test script* yang berhasil dieksekusi seperti pada gambar 5-12.

```

Time of Emulation Session: Sun Dec 21 2008 14:25

                                Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:                      0.0 secs  Inactive Time:                      0.0 secs

Commands:                          0      Average Throughput: UNDEFN cmds/min
Passed Responses:                   0      Average Throughput: UNDEFN resps/min
Failed Responses:                   0      Average Throughput: UNDEFN resps/min

Time Spent Waiting:                 0.0 seconds => UNDEFN
Time Spent in Input:                0.0 seconds => UNDEFN
Time Spent Thinking:                0.0 seconds => UNDEFN

Summary Statistics

Duration of Run:                    0.0 seconds (00:00:00.0)

Summary Statistics for Selected Virtual Testers and Scripts:

Commands:                          0      Total Throughput: UNDEFN cmds/min

Number of Virtual Testers:          1      Number of stop_time Cmds:          0
Number of Completed Scripts:        11     Number of Uncompleted Scripts:     0

Average Number of Scripts Completed per Virtual Tester: 11.0 scripts
Average Duration for Completed Scripts: 0.0 seconds
Script Throughput for Completed Scripts: UNDEFN scripts/hour

```

Gambar 5-12. Command usage report uji coba fungsi SIMBION final 3

- Uji coba fungsi SIMBION final 4

Seluruh pengujian berhasil. Hal tersebut dapat terlihat pada gambar 5-13.

Event Type	Result	Date & Time	Failure R
[-] Suite Start (Uji coba fungsi SIMBION 4)	Pass	12/21/2008 2:36:14 ...	
[-] Computer Start (Suite 1 [1])	Warning	12/21/2008 2:36:15 ...	
[+] Script Start (Log in)	Pass	12/21/2008 2:36:17 ...	
[+] Script Start (Profil)	Pass	12/21/2008 2:36:20 ...	
[+] Script Start (Daftar beasiswa)	Pass	12/21/2008 2:36:51 ...	
[+] Script Start (Daftar Pengajuan Beasiswa)	Pass	12/21/2008 2:36:51 ...	
[+] Script Start (Lihat detail daftar pengajuan bea...)	Pass	12/21/2008 2:37:23 ...	
[+] Script Start (Kembali ke detail daftar pengaju...)	Pass	12/21/2008 2:37:23 ...	
[+] Script Start (Log out)	Pass	12/21/2008 2:37:56 ...	
Computer End	Warning	12/21/2008 2:37:56 ...	
[-] Suite End (Uji coba fungsi SIMBION 4)	Pass	12/21/2008 2:38:00 ...	

Gambar 5-13. Test log uji coba fungsi SIMBION final 4

Command data report dari pengujian ini dapat dilihat pada gambar 5-14.

```
==> Start of data for Virtual Tester: Suite 1[1] <==
==> End of data for Virtual Tester: Suite 1[1] <==
```

Gambar 5-14. Command data report uji coba fungsi SIMBION final 4

Time stamp pengujian dapat terlihat pada gambar 5-15.

```
Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 1
Number of Virtual Testers in Report: 1

Information for Virtual Tester: Suite 1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 14:36:14 2008
Time of Day Reference for Timestamps: 1229844974

Virtual Tester: Suite 1[1]          Test Script: Log in
Beginning timestamp of test script Log in:      3749
Beginning timestamp of test script Profil:      6046
Beginning timestamp of test script Daftar beasiswa: 37124
Beginning timestamp of test script Daftar Pengajuan Beasiswa: 37984
Beginning timestamp of test script Lihat detil daftar pengajuan beasiswa2: 69125
Beginning timestamp of test script Kembali ke detil daftar pengajuan2: 69983
Beginning timestamp of test script Log out:     102422
```

Gambar 5-15. Command trace report uji coba fungsi SIMBION final 4

Pada command usage report melaporkan bahwa ada tujuh script yang dijalankan. Hal tersebut dapat dilihat pada gambar 5-16.

```
Time of Emulation Session: Sun Dec 21 2008 14:36

Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:          0.0 secs  Inactive Time:          0.0 secs

Commands:             0          Average Throughput: UNDEFN cmds/min
Passed Responses:    0          Average Throughput: UNDEFN resps/min
Failed Responses:    0          Average Throughput: UNDEFN resps/min

Time Spent Waiting:   0.0 seconds => UNDEFN
Time Spent in Input:  0.0 seconds => UNDEFN
Time Spent Thinking:  0.0 seconds => UNDEFN

Summary Statistics

Duration of Run:      0.0 seconds (00:00:00.0)

Summary Statistics for Selected Virtual Testers and Scripts:

Commands:             0          Total Throughput: UNDEFN cmds/min

Number of Virtual Testers: 1      Number of stop_time Cmds: 0
Number of Completed Scripts: 7      Number of Uncompleted Scripts: 0

Average Number of Scripts Completed per Virtual Tester: 7.0 scripts
Average Duration for Completed Scripts: 0.0 seconds
Script Throughput for Completed Scripts: UNDEFN scripts/hour
```

Gambar 5-16. Command usage report uji coba fungsi SIMBION final 4

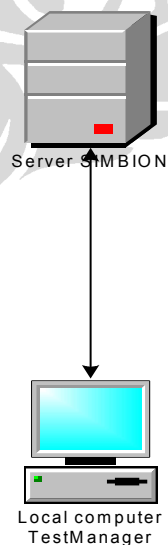
5.4 *Functional testing* pada lebih dari satu komputer

Pada bagian ini akan dilakukan pengujian yang sama. Namun, pengujian ini dilakukan pada komputer *agent*. *Local computer* yang ada akan berlaku sebagai *server* dan komputer lainnya sebagai *agent* atau *virtual tester*. Spesifikasi komputer yang digunakan untuk pengujian adalah sebagai berikut:

- *Local computer*:
 - Prosesor: Pentium 4
 - Memory: 2 GB
 - System operasi: Windows XP SP 2
- Komputer 1:

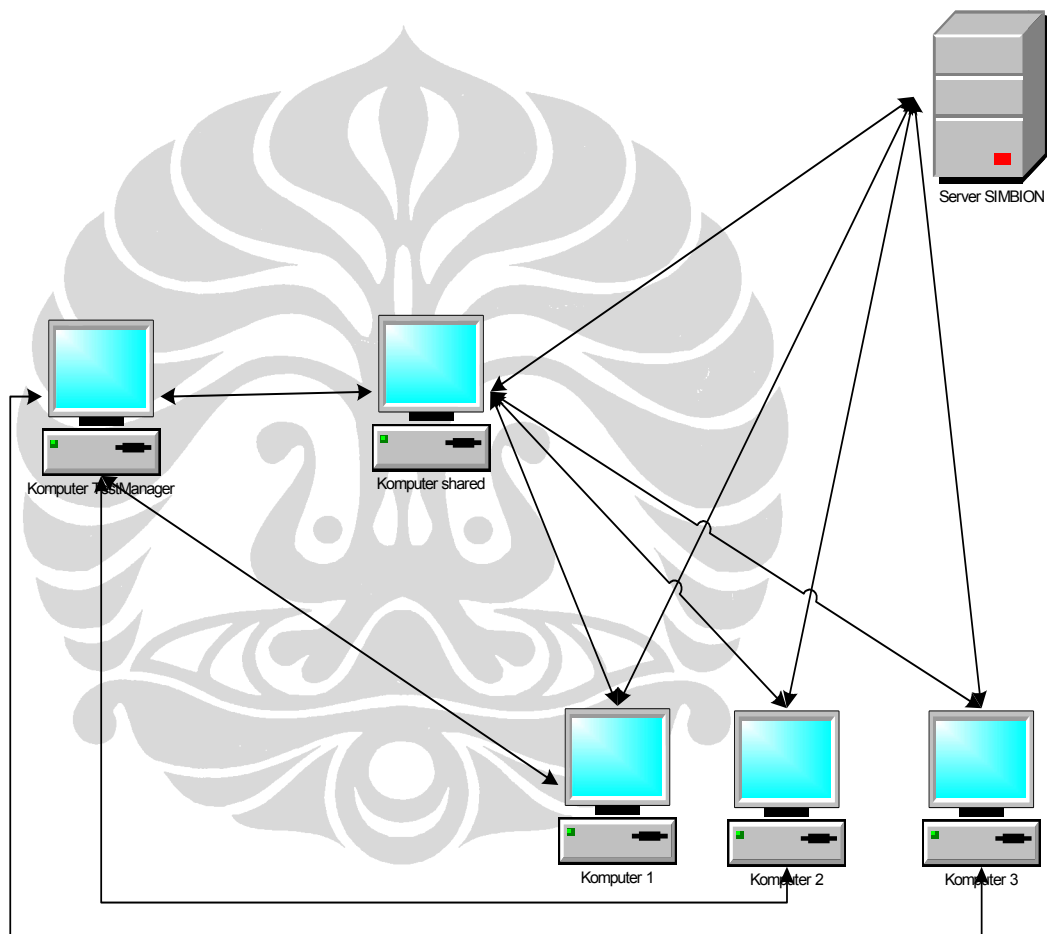
- Prosesor: Pentium 4
- Memory: 512 MB
- System operasi: Windows XP SP 2
- Komputer 2:
 - Prosesor: Pentium 4
 - Memory: 256 MB
 - System operasi: Windows XP SP 2
- Komputer 3:
 - Prosesor: Pentium 4
 - Memory: 256 MB
 - System operasi: Windows XP SP 2
- Komputer shared:
 - Prosesor: Intel Core 2 Duo
 - Memory: 2 GB
 - System operasi: Windows XP SP 2

Pada pengujian ini, arsitektur pengujian akan berbeda dengan pengujian yang dilakukan pada *Local computer*. Pada pengujian di *Local computer*, komputer yang melakukan pengujian dengan menjalankan Rational TestManager akan terhubung langsung pada *server* SIMBION.



Gambar 5-17. Arsitektur local computer

Pada pengujian dengan *virtual tester*, bentuk arsitektur akan berbeda. *Local computer* sebagai TestManager dan komputer *agent* tidak dapat langsung melakukan pengujian ke *server* SIMBION. Folder dari proyek pengujian tidak boleh berada di komputer yang sama dengan *Local computer*, komputer *agent* maupun *server*. Oleh karena itu, sebuah komputer harus dialokasikan sebagai komputer *shared* tempat proyek pengujian berada. *Local computer*, *server* dan komputer *agent* akan terhubung langsung ke komputer *shared*.



Gambar 5-18. Arsitektur virtual tester

Ada beberapa hal yang harus diperhatikan sebelum melakukan pengujian, yaitu:

- Seluruh komputer yang ada, kecuali komputer *shared* harus terinstal *software* Rational dengan versi yang sama.
- Internet Explorer yang ada pada setiap komputer harus versi 6.

- Pada setiap komputer *agent* menjalankan Rational Test *Agent*.
- *Firewall* pada setiap komputer tidak boleh aktif.
- Pada saat menjalankan pengujian sebaiknya hanya aplikasi Rational saja yang berjalan.
- Tidak menggunakan *proxy* internet.

Proses perencanaan hingga pembuatan *test suite* sama dengan proses pengujian pada *Local computer*. Untuk melakukan pengujian pada komputer *agent* ada beberapa hal yang harus dilakukan:

- Mengalokasikan *test suite* yang akan diuji untuk bisa dijalankan oleh lebih dari satu komputer.
- Mengatur urutan komputer yang akan melakukan pengujian. Pengujian akan dilakukan secara urut sesuai nomor.

Pada saat melakukan pengujian, setiap *test script* dapat langsung diatur untuk diuji oleh lebih dari satu komputer. Pada bagian ini *test suite* yang akan diuji adalah *test script* Uji coba fungsi SIMBION final. *Test script* tersebut mewakili *test case* keempat pada bab sebelumnya. Hasil dari pengujian adalah sebagai berikut:

- Hampir sebagian *test script* gagal.
- Komputer *agent* yang berjalan ketika diuji hanya satu, yaitu komputer *agent* dengan urutan paling atas.

Setelah *test suite* selesai dijalankan, akan muncul laporan pengujian. Berikut adalah laporan hasil pengujian:

Pada *test log* yang ada memperlihatkan bahwa keseluruhan *test suite* gagal, tetapi ada *test script* yang berhasil. *Test log* dapat dilihat pada gambar 5-19.

Event Type	Result	Date & Time	Failure R
Suite Start (Fungsi simbion)	Fail	12/21/2008 6:52:30 ...	Executabl
Computer Start (Suite 1 [1])	Fail	12/21/2008 6:50:19 ...	
+ Script Start (Log in)	Pass	12/21/2008 6:50:20 ...	
+ Script Start (Profil)	Pass	12/21/2008 6:50:22 ...	
+ Script Start (Ubah profil)	Fail	12/21/2008 6:50:23 ...	
Message	Informational	12/21/2008 6:50:38 ...	
+ Script Start (Load SIAK)	Fail	12/21/2008 6:50:38 ...	
Message	Informational	12/21/2008 6:51:23 ...	
+ Script Start (Simpan profil)	Fail	12/21/2008 6:51:23 ...	
Message	Informational	12/21/2008 6:51:53 ...	
+ Script Start (Daftar beasiswa)	Pass	12/21/2008 6:51:54 ...	
+ Script Start (Detil beasiswa)	Pass	12/21/2008 6:51:54 ...	
+ Script Start (Ajukan beasiswa)	Fail	12/21/2008 6:52:26 ...	
Message	Informational	12/21/2008 6:52:26 ...	
+ Script Start (Simpan pengajuan beasiswa)	Fail	12/21/2008 6:52:27 ...	
Message	Informational	12/21/2008 6:53:27 ...	
+ Script Start (Detil daftar pengajuan)	Fail	12/21/2008 6:53:27 ...	
Message	Informational	12/21/2008 6:53:57 ...	
+ Script Start (Ubah detil daftar pengajuan)	Fail	12/21/2008 6:53:57 ...	
Message	Informational	12/21/2008 6:54:28 ...	
+ Script Start (Simpan ubah detil pengejuan)	Fail	12/21/2008 6:54:28 ...	
Message	Informational	12/21/2008 6:54:58 ...	
+ Script Start (Log out)	Pass	12/21/2008 6:54:58 ...	
Computer End	Fail	12/21/2008 6:54:59 ...	
Suite End (Fungsi simbion)	Fail	12/21/2008 6:57:14 ...	Executabl

Gambar 5-19. Test log pengujian dengan virtual tester

Pada *command trace report* terlihat bahwa waktu untuk melakukan *test script* yang gagal lebih lama, karena TestManager mencoba berulang kali terlebih dahulu sebelum menyatakan gagal. *Command trace report* dapat dilihat pada gambar 5-20.

```
Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 1
Number of Virtual Testers in Report: 1

Information for Virtual Tester: Suite 1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 18:50:18 2008
Time of Day Reference for Timestamps: 1229860218

Virtual Tester: Suite 1[1]          Test Script: Log in
Beginning timestamp of test script Log in:      2062
Beginning timestamp of test script Profil:      4734
Beginning timestamp of test script Ubah profil:  5687
Beginning timestamp of test script Load SIAK:  20436
Beginning timestamp of test script Simpan profil: 65202
Beginning timestamp of test script Daftar beasiswa: 96015
Beginning timestamp of test script Detil beasiswa: 96936
Beginning timestamp of test script Ajukan beasiswa: 128233
Beginning timestamp of test script Simpan pengajuan beasiswa: 129000
Beginning timestamp of test script Detil daftar pengajuan: 189608
Beginning timestamp of test script Ubah detil daftar pengajuan: 219952
Beginning timestamp of test script Simpan ubah detil pengejuan: 250311
Beginning timestamp of test script Log out:     280655
```

Gambar 5-20. Command trace report pengujian dengan virtual tester

Command usage report menunjukkan bahwa seluruh *test script* telah dieksekusi seperti pada gambar 5-21.

```

Time of Emulation Session: Sun Dec 21 2008 18:50

                          Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:                0.0 secs  Inactive Time:                0.0 secs
Commands:                   0          Average Throughput: UNDEFN cmds/min
Passed Responses:           0          Average Throughput: UNDEFN resps/min
Failed Responses:           0          Average Throughput: UNDEFN resps/min

Time Spent Waiting:         0.0 seconds => UNDEFN
Time Spent in Input:        0.0 seconds => UNDEFN
Time Spent Thinking:        0.0 seconds => UNDEFN

Summary Statistics

Duration of Run:             0.0 seconds (00:00:00.0)

Summary Statistics for Selected Virtual Testers and Scripts:

Commands:                   0          Total Throughput: UNDEFN cmds/min

Number of Virtual Testers:   1          Number of stop_time Cnds:      0
Number of Completed Scripts: 13         Number of Uncompleted Scripts: 0

Average Number of Scripts Completed per Virtual Tester: 13.0 scripts
Average Duration for Completed Scripts: 0.0 seconds
Script Throughput for Completed Scripts: UNDEFN scripts/hour

```

Gambar 5-21. Command usage report pengujian dengan virtual tester

Pada pengujian dengan *virtual tester* ini, ada beberapa masalah yang ditemui, yaitu:

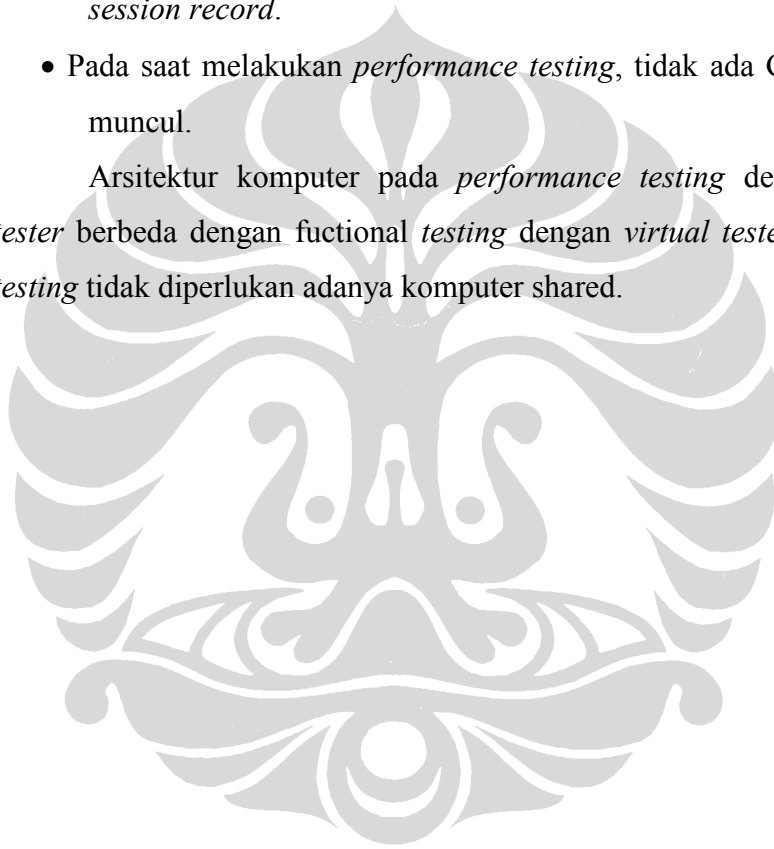
- Untuk *test suite* dengan jumlah *test script* yang sedikit dan sederhana, pengujian akan berhasil sepenuhnya.
- Apabila *test suite* yang diuji memiliki jumlah *test script* lebih dari dua dan kompleks, hasil pengujian tidak akan 100% berhasil.
- Pada setiap pengujian, hanya ada satu komputer yang menguji, meskipun pengujian sudah dialokasikan pada lebih dari satu komputer.

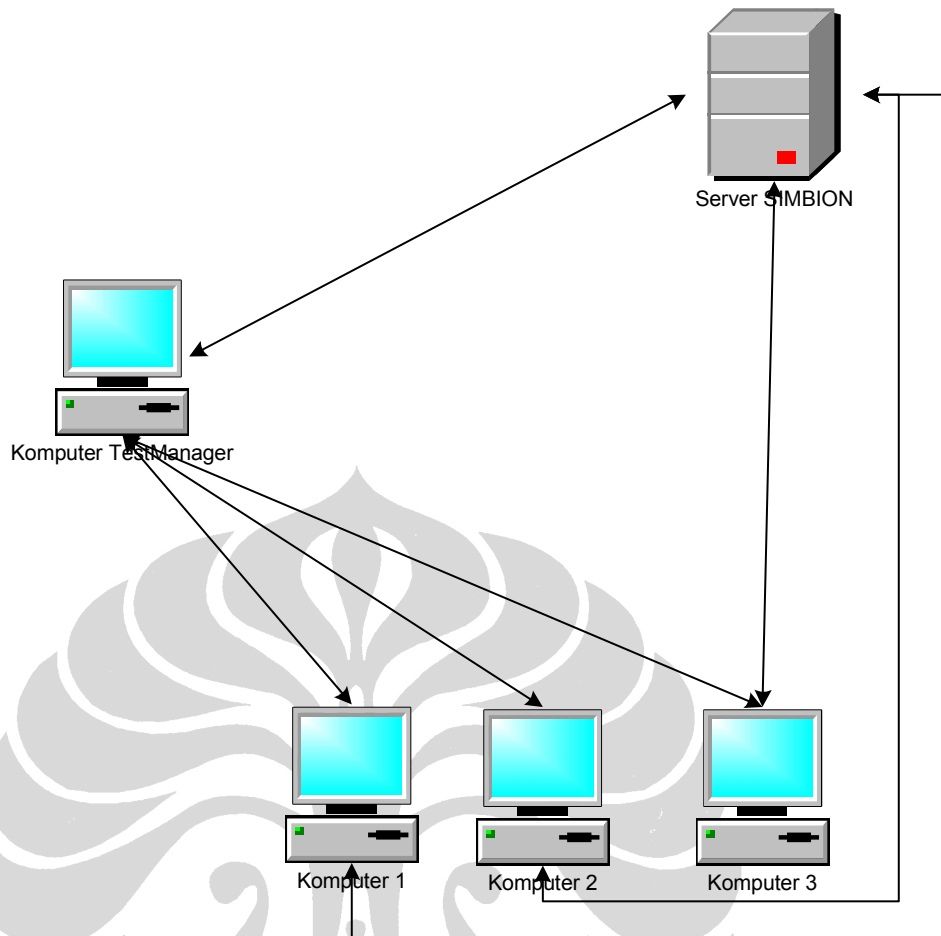
5.5 *Performance testing*

Pada bagian ini akan dipaparkan bagaimana melakukan *performance testing* pada SIMBION. *Test plan* yang digunakan pada *performance testing* sama dengan *test plan* sebelumnya. Ada beberapa perbedaan antara *functional testing* dengan *performance testing*, yaitu:

- Pada *performance testing* Rational Robot merekam *session*, sedangkan pada *functional testing* merekam GUI. Proses perekaman mirip dengan proses perekaman pada GUI, hanya saja *performance testing* menggunakan *session record*.
- Pada saat melakukan *performance testing*, tidak ada GUI dari sistem yang muncul.

Arsitektur komputer pada *performance testing* dengan banyak *virtual tester* berbeda dengan *functional testing* dengan *virtual tester*. Pada *performance testing* tidak diperlukan adanya komputer shared.



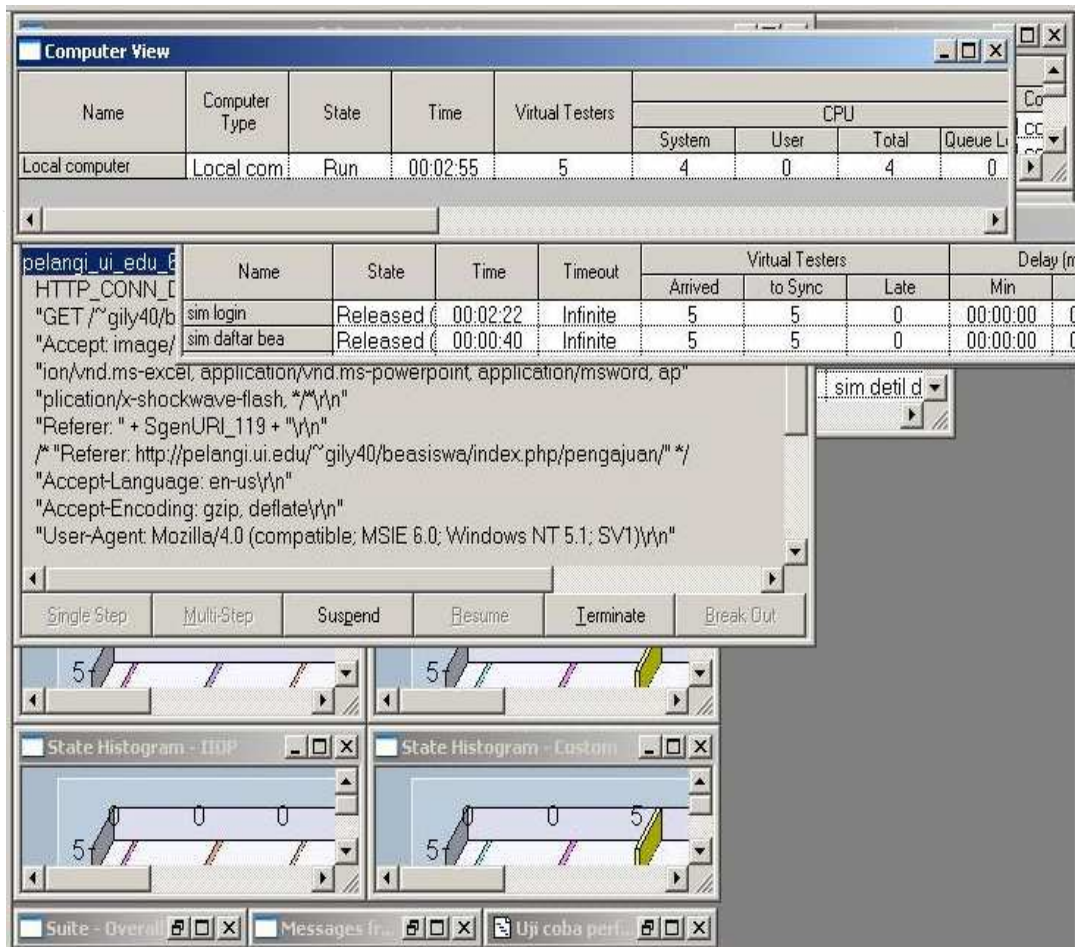


Gambar 5-22. Arsitektur performance testing

Performance testing yang akan dilakukan akan menguji *test script* Uji coba performance SIMBION 4 yang berdasarkan *test script* Uji coba fungsi SIMBION final 4 pada *functional testing* sebelumnya. sebelum melakukan pengujian *test script* yang akan diuji juga dialokasikan dengan banyak komputer dan virtual user. Pada satu komputer dapat dibuat lebih dari satu *virtual user*. Spesifikasi komputer yang akan digunakan sama dengan pengujian sebelumnya.

Pada proses pengujian, setiap komputer dan *virtual user* yang telah dialokasikan akan menguji setiap *session* yang direkam. Berikut adalah hasil pengujian menggunakan *Local computer* dengan jumlah *virtual user* sebanyak 5 buah:

Pada gambar 5-23 memperlihatkan proses pengujian yang telah dilakukan.



Gambar 5-23. Proses performance testing

Hasil dari performance testing berhasil seluruhnya. Status untuk setiap *test script* adalah *Pass*. *Test log* dapat dilihat pada gambar 5-24.

Event Type	Result	Date & Time	Failure R
[-] Suite Start (Uji coba performance SIMBION 4)	Pass	12/21/2008 4:33:41 ...	
[-] User Start (VU User Group1 [1])	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
Env Variable Change	Pass	12/21/2008 4:33:56 ...	
+ Script Start (sim buka browser)	Pass	12/21/2008 4:33:56 ...	
+ Script Start (sim login)	Pass	12/21/2008 4:33:57 ...	
+ Script Start (sim profil)	Pass	12/21/2008 4:34:31 ...	
+ Script Start (sim ubah profil)	Pass	12/21/2008 4:35:18 ...	
+ Script Start (sim simpan profil)	Pass	12/21/2008 4:35:38 ...	
+ Script Start (sim daftar bea)	Pass	12/21/2008 4:35:58 ...	
+ Script Start (sim daftar pengajuan)	Pass	12/21/2008 4:36:12 ...	
+ Script Start (sim detil daftar pengajuan)	Pass	12/21/2008 4:36:40 ...	
+ Script Start (sim kembali ke detil daftar)	Pass	12/21/2008 4:37:16 ...	
+ Script Start (sim log out)	Pass	12/21/2008 4:37:34 ...	
+ Script Start (sim selesai)	Pass	12/21/2008 4:37:48 ...	
User End	Pass	12/21/2008 4:37:48 ...	
+ User Start (VU User Group1 [2])	Pass	12/21/2008 4:33:56 ...	
+ User Start (VU User Group1 [3])	Pass	12/21/2008 4:33:56 ...	
+ User Start (VU User Group1 [4])	Pass	12/21/2008 4:33:56 ...	
+ User Start (VU User Group1 [5])	Pass	12/21/2008 4:33:57 ...	
Suite End (Uji coba performance SIMBION 4)	Pass	12/21/2008 4:37:48 ...	

Gambar 5-24. Test log performance testing 5 virtual user

Command data report menunjukkan ada 5 *virtual tester* yang dijalankan secara berurutan. Hal tersebut dapat dilihat pada gambar 5-25.

```

==> Start of data for Virtual Tester: WU User Group1[1] <==
==> End of data for Virtual Tester: WU User Group1[1] <==
==> Start of data for Virtual Tester: WU User Group1[2] <==
==> End of data for Virtual Tester: WU User Group1[2] <==
==> Start of data for Virtual Tester: WU User Group1[3] <==
==> End of data for Virtual Tester: WU User Group1[3] <==
==> Start of data for Virtual Tester: WU User Group1[4] <==
==> End of data for Virtual Tester: WU User Group1[4] <==
==> Start of data for Virtual Tester: WU User Group1[5] <==
==> End of data for Virtual Tester: WU User Group1[5] <==

```

Gambar 5-25. Command data report performance testing 5 virtual user

Command trace report menunjukkan *timestamp* dari pengujian. Dapat terlihat juga detail dari perintah yang dijalankan dengan status hasilnya. *Command trace report* dapat dilihat pada gambar 5-26.

```

Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 5
Number of Virtual Testers in Report: 5

Information for Virtual Tester: WU User Group1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 16:33:41 2008
Time of Day Reference for Timestamps: 1229852021

Virtual Tester's Environment: Screen = 0, Server_connection = 1, Emulation = NONE,
                             Screen_match = CURSOR_DATA

Login Timestamp:      15735

Virtual Tester: WU User Group1[1]      Test Script: sim buka browser

Beginning timestamp of test script sim buka browser:      15735

Src  Cmd
Line Count Clnt Command      Command ID      Count      First      Last
-----
31   1   1 http_request sim buk~1.001   279        15735     15735     15735     pass

Virtual Tester's Environment: Server_connection = 1 ("sim buk~1.001")

47   2   1 http_hdr_rcv sim buk~1.002   319        15844     15844     15844     pass
49   3   1 http_nrcv   sim buk~1.003   4507       15844     15844     15844     pass
54   4   2 http_request sim buk~1.004   405        16172     16172     16172     pass

Virtual Tester's Environment: Server_connection = 2 ("sim buk~1.004")

71   5   2 http_hdr_rcv sim buk~1.005   211        16172     16172     16172     pass
73   6   2 http_nrcv_cache sim buk~1.006   0          16172     16172     16172     pass
78   7   2 http_request sim buk~1.007   200        16172     16172     16172     pass

```

Gambar 5-26. Command trace report performance testing 5 virtual user

Pada *command usage report* dapat terlihat bahwa jumlah perintah untuk setiap *virtual tester* adalah 140. Pada gambar 5-27 dapat juga dilihat waktu yang

dibutuhkan oleh setiap *virtual tester*.

```
Time of Emulation Session: Sun Dec 21 2008 16:33

Command Usage Report

Cumulative Statistics for All Selected Virtual Testers and Scripts

Active Time:          1157.3 secs  Inactive Time:          79.6 secs

Passed Commands:      140          Average Throughput:     7.26 cmds/min
Failed Commands:       0          Average Throughput:     0.00 cmds/min
Passed Responses:     140          Average Throughput:     7.26 resps/min
Failed Responses:      0          Average Throughput:     0.00 resps/min

Time Spent Waiting:   7.7 seconds => 0.7%
Time Spent in Input:  0.0 seconds => 0.0%
Time Spent Thinking: 1149.6 seconds => 99.3%

Passed HTTP Connections: 140      Passed Connections/Min: 7.3
Failed HTTP Connections: 0          Avg Connect Setup Time: 0 ms
HTTP Sent Kbytes:       70          HTTP Sent KBytes/Sec:  0.1
HTTP Received Kbytes:   387         HTTP Recv KBytes/Sec:  0.3
Sent Kbytes/Connection: 0.5        Recv Kbytes/Connection: 2.8

stop_time Commands:    10          start_time Commands:    10
stop_time Cnds/Minute: 0.52        Avg Block Time:         0.000 secs

Summary Statistics

Duration of Run:       231.8 seconds (00:03:51.8)

Summary Statistics for Selected Virtual Testers and Scripts:

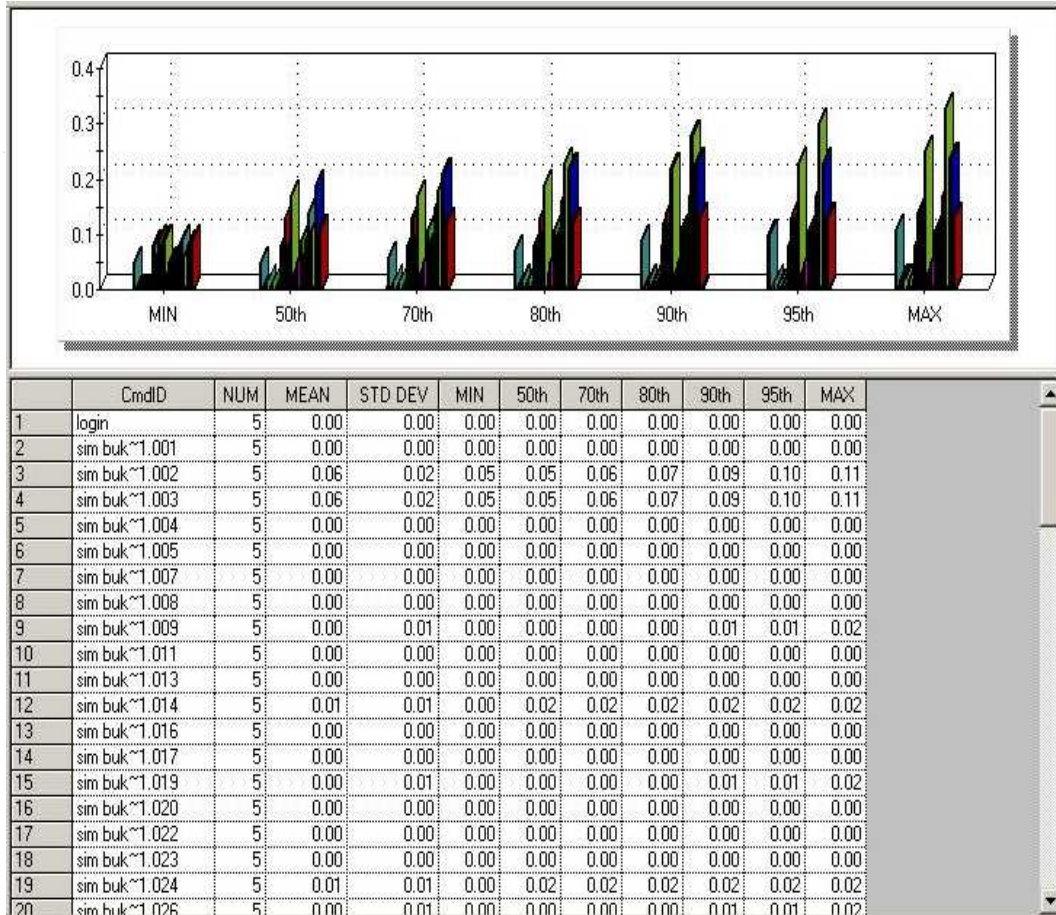
Passed Commands:      140          Total Throughput:     36.24 cmds/min
Failed Commands:       0          Total Throughput:     0.00 cmds/min
Passed Responses:     140          Total Throughput:     36.24 resps/min
Failed Responses:      0          Total Throughput:     0.00 resps/min

Number of Virtual Testers: 5          Number of stop_time Cnds: 10
```

Gambar 5-27. Command usage report performance testing 5 virtual user

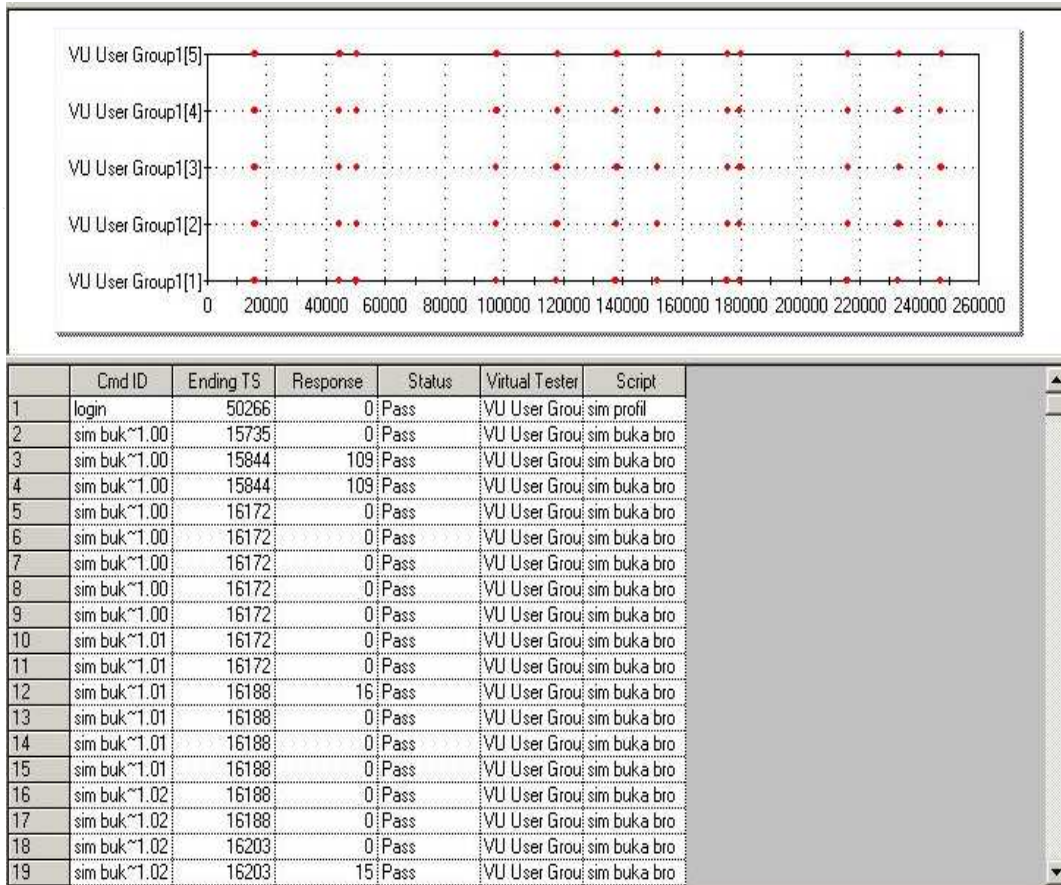
Salah satu hasil dari pengujian adalah *performance report*. Pada laporan tersebut terlihat diagram yang menunjukkan performa system ketika diuji oleh 5 *virtual*

user. Performance report dapat dilihat pada gambar 5-28.



Gambar 5-28. Performance report dari performance testing 5 virtual user

Pada *response vs. time report* dapat terlihat laporan respon waktu dari masing-masing script. *Response vs. time report* dapat dilihat pada gambar 5-29.



Gambar 5-29. Response vs time report dari performance testing 5 virtual user

Berikut adalah hasil pegujian pada dua komputer, yaitu komputer 1 dan komputer 2 dengan jumlah *virtual user* 100.

Hasil dari performance testing berhasil seluruhnya. Status untuk setiap *test script* adalah *Pass*. *Test log* dapat dilihat pada gambar 5-30.

Event Type	Result	Date & Time	Failure
[-] Suite Start (Uji coba performance SIMBION 4)	Pass	12/21/2008 4:45:46 ...	
+ User Start (VU User Group1 [1])	Pass	12/21/2008 9:41:17 ...	
+ User Start (VU User Group1 [3])	Pass	12/21/2008 9:41:17 ...	
+ User Start (VU User Group1 [5])	Pass	12/21/2008 9:41:17 ...	
+ User Start (VU User Group1 [7])	Pass	12/21/2008 9:41:17 ...	
+ User Start (VU User Group1 [9])	Pass	12/21/2008 9:41:18 ...	
+ User Start (VU User Group1 [11])	Pass	12/21/2008 9:41:18 ...	
+ User Start (VU User Group1 [13])	Pass	12/21/2008 9:41:18 ...	
+ User Start (VU User Group1 [15])	Pass	12/21/2008 9:41:18 ...	
+ User Start (VU User Group1 [17])	Pass	12/21/2008 9:41:18 ...	
+ User Start (VU User Group1 [19])	Pass	12/21/2008 9:41:19 ...	
+ User Start (VU User Group1 [21])	Pass	12/21/2008 9:41:19 ...	
+ User Start (VU User Group1 [23])	Pass	12/21/2008 9:41:19 ...	
+ User Start (VU User Group1 [25])	Pass	12/21/2008 9:41:19 ...	
+ User Start (VU User Group1 [27])	Pass	12/21/2008 9:41:20 ...	
+ User Start (VU User Group1 [29])	Pass	12/21/2008 9:41:20 ...	
+ User Start (VU User Group1 [31])	Pass	12/21/2008 9:41:20 ...	
+ User Start (VU User Group1 [33])	Pass	12/21/2008 9:41:20 ...	
+ User Start (VU User Group1 [35])	Pass	12/21/2008 9:41:20 ...	
+ User Start (VU User Group1 [37])	Pass	12/21/2008 9:41:21 ...	
+ User Start (VU User Group1 [39])	Pass	12/21/2008 9:41:21 ...	
+ User Start (VU User Group1 [41])	Pass	12/21/2008 9:41:21 ...	
+ User Start (VU User Group1 [43])	Pass	12/21/2008 9:41:21 ...	
+ User Start (VU User Group1 [45])	Pass	12/21/2008 9:41:21 ...	
+ User Start (VU User Group1 [47])	Pass	12/21/2008 9:41:22 ...	
+ User Start (VU User Group1 [49])	Pass	12/21/2008 9:41:22 ...	
+ User Start (VU User Group1 [51])	Pass	12/21/2008 9:41:22 ...	
+ User Start (VU User Group1 [53])	Pass	12/21/2008 9:41:22 ...	
+ User Start (VU User Group1 [55])	Pass	12/21/2008 9:41:22 ...	

Gambar 5-30. Test log performance testing 100 virtual user

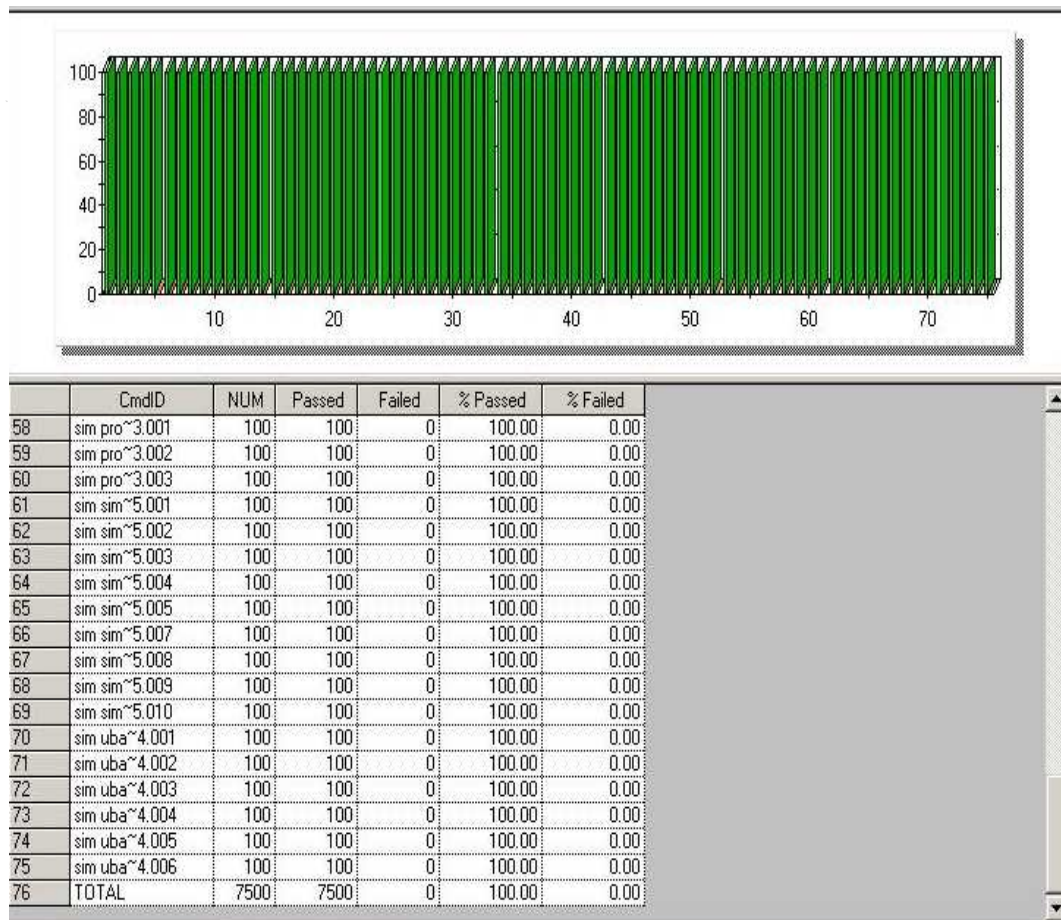
Command data report menunjukkan ada 100 *virtual tester* yang dijalankan secara berurutan. Hal tersebut dapat dilihat pada gambar 5-31.

```
==> Start of data for Virtual Tester: VU User Group1[1] <==
==> End of data for Virtual Tester: VU User Group1[1] <==
==> Start of data for Virtual Tester: VU User Group1[3] <==
==> End of data for Virtual Tester: VU User Group1[3] <==
==> Start of data for Virtual Tester: VU User Group1[5] <==
==> End of data for Virtual Tester: VU User Group1[5] <==
==> Start of data for Virtual Tester: VU User Group1[7] <==
==> End of data for Virtual Tester: VU User Group1[7] <==
==> Start of data for Virtual Tester: VU User Group1[9] <==
==> End of data for Virtual Tester: VU User Group1[9] <==
==> Start of data for Virtual Tester: VU User Group1[11] <==
==> End of data for Virtual Tester: VU User Group1[11] <==
==> Start of data for Virtual Tester: VU User Group1[13] <==
==> End of data for Virtual Tester: VU User Group1[13] <==
==> Start of data for Virtual Tester: VU User Group1[15] <==
==> End of data for Virtual Tester: VU User Group1[15] <==
==> Start of data for Virtual Tester: VU User Group1[17] <==
==> End of data for Virtual Tester: VU User Group1[17] <==
==> Start of data for Virtual Tester: VU User Group1[19] <==
==> End of data for Virtual Tester: VU User Group1[19] <==
==> Start of data for Virtual Tester: VU User Group1[21] <==
==> End of data for Virtual Tester: VU User Group1[21] <==
==> Start of data for Virtual Tester: VU User Group1[23] <==
==> End of data for Virtual Tester: VU User Group1[23] <==
==> Start of data for Virtual Tester: VU User Group1[25] <==
==> End of data for Virtual Tester: VU User Group1[25] <==
==> Start of data for Virtual Tester: VU User Group1[27] <==
==> End of data for Virtual Tester: VU User Group1[27] <==
==> Start of data for Virtual Tester: VU User Group1[29] <==
==> End of data for Virtual Tester: VU User Group1[29] <==
==> Start of data for Virtual Tester: VU User Group1[31] <==
==> End of data for Virtual Tester: VU User Group1[31] <==
==> Start of data for Virtual Tester: VU User Group1[33] <==
==> End of data for Virtual Tester: VU User Group1[33] <==
==> Start of data for Virtual Tester: VU User Group1[35] <==
==> End of data for Virtual Tester: VU User Group1[35] <==
==> Start of data for Virtual Tester: VU User Group1[37] <==
==> End of data for Virtual Tester: VU User Group1[37] <==
==> Start of data for Virtual Tester: VU User Group1[39] <==
==> End of data for Virtual Tester: VU User Group1[39] <==
==> Start of data for Virtual Tester: VU User Group1[41] <==
```

Gambar 5-31. Command data report performance testing 100 virtual user

Command status report menggambarkan status dari setiap *script* yang dijalankan. Status juga ditunjukkan dalam diagram. Jika berwarna hijau maka statusnya

adalah *Pass*, sedangkan merah berarti *failed*. *Command status report* dapat dilihat pada gambar 5-32.



Gambar 5-32. Command status report dari performance testing 100 virtual user

Command trace report menunjukkan *time stamp* dari pengujian. Dapat terlihat juga detail dari perintah yang dijalankan dengan status hasilnya. *Command trace report* melaporkan seluruh *test script* yang dijalankan seperti terlihat pada gambar 5-33.

```

Trace of Emulation Session Activity

Total Number of Virtual Testers Emulated: 100
Number of Virtual Testers in Report: 100

Information for Virtual Tester: VU User Group1[1]
Rational Suite TestStudio Release 8.5

Time of Suite Run: Sun Dec 21 09:41:11 2008
Time of Day Reference for Timestamps: 1229827271

Virtual Tester's Environment: Screen = 0, Server_connection = 1, Emulation = NONE,
                             Screen_match = CURSOR_DATA

Login Timestamp:          6516

Virtual Tester: VU User Group1[1]          Test Script: sim buka browser

Beginning timestamp of test script sim buka browser:          6516

Src  Cmd
Line Count Clnt Command          Command ID          Count          First          Last
-----
31   1   1 http_request sim buk~1.001      279           6516           6516
                                     6516           6516 pass

Virtual Tester's Environment: Server_connection = 1 ("sim buk~1.001")

47   2   1 http_hdr_rcv sim buk~1.002      319           6625           6625 pass
49   3   1 http_nrcv   sim buk~1.003      4507          6625           6625 pass
54   4   2 http_request sim buk~1.004      405           6952           6952
                                     6952           6952 pass

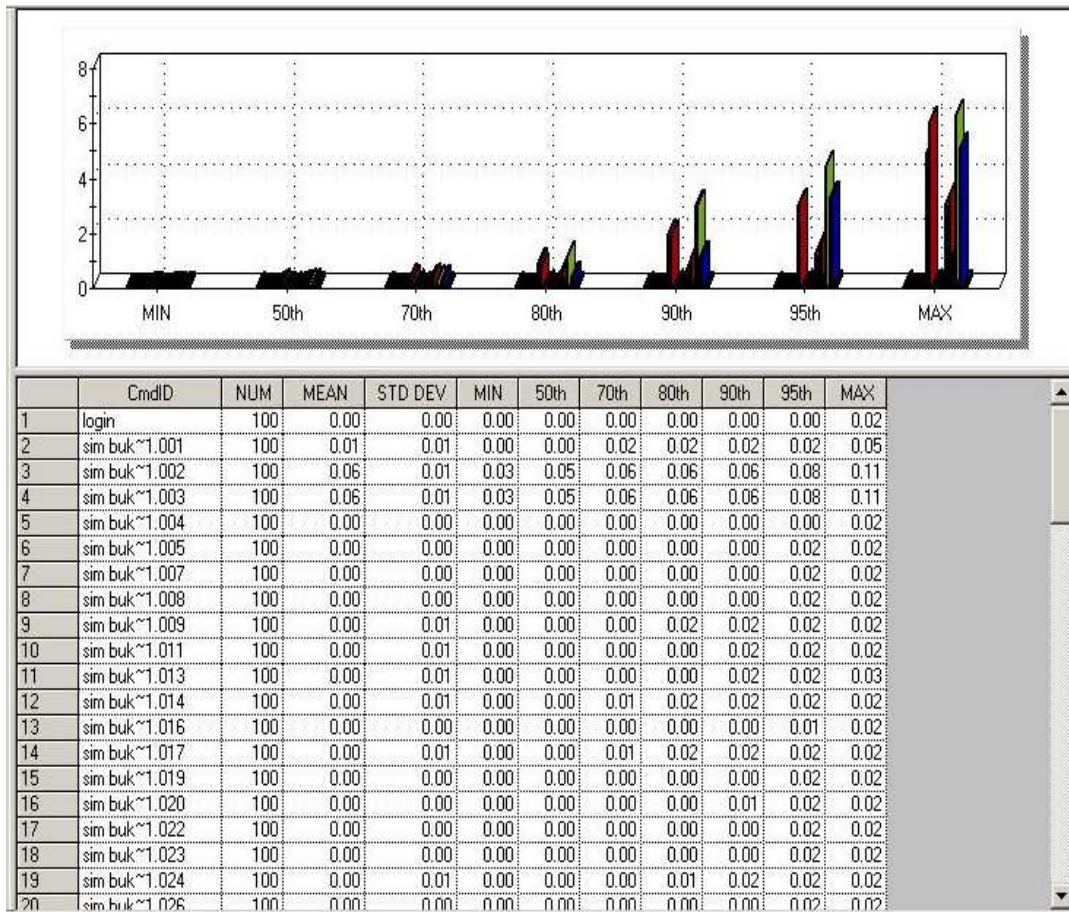
Virtual Tester's Environment: Server_connection = 2 ("sim buk~1.004")

71   5   2 http_hdr_rcv sim buk~1.005      211           6952           6952 pass
73   6   2 http_nrcv_cache sim buk~1.006      0             6952           6952 pass
79   7   2 http_request sim buk~1.007      200           6952           6952

```

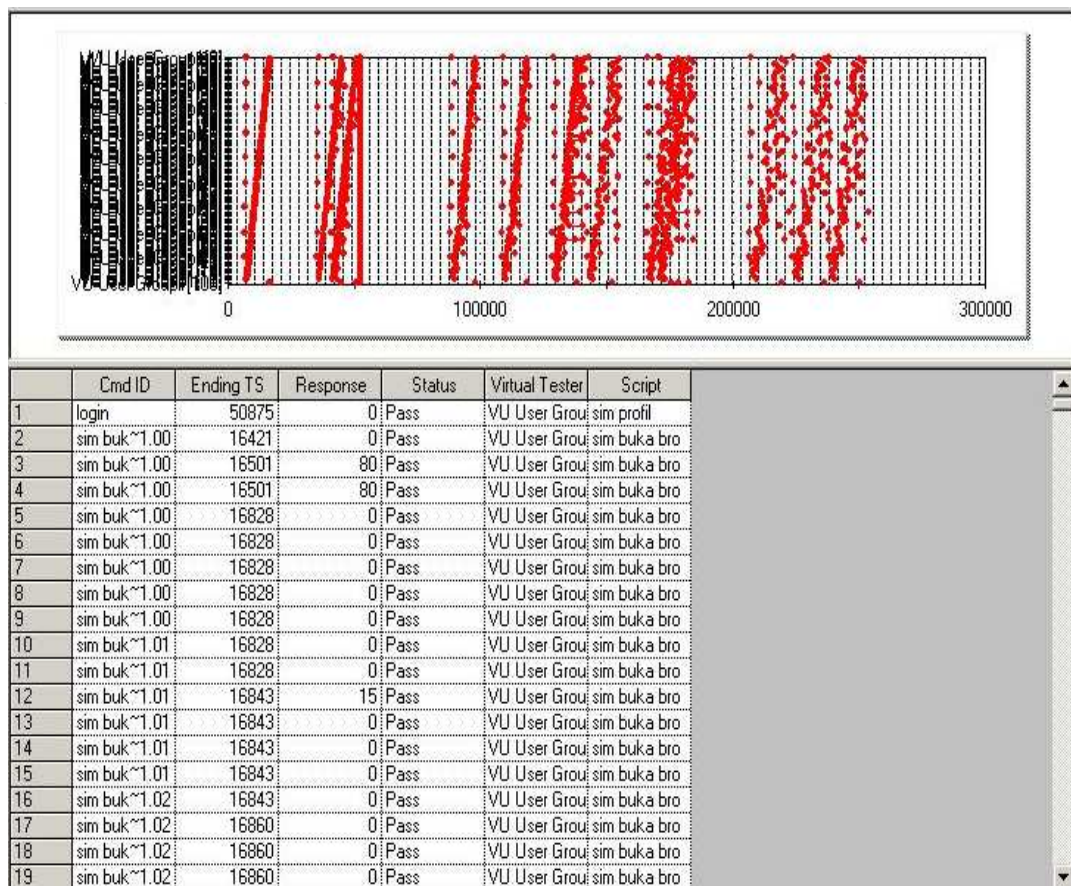
Gambar 5-33. Command trace report dari performance testing 100 virtual user

Pada performance *report* tersebut terlihat diagram yang menunjukkan performa sistem ketika diuji oleh 100 *virtual user*. *Performance report* dapat dilihat pada gambar 5-34.



Gambar 5-34. Performance report dari performance testing 100 virtual user

Pada *response vs. time report* dapat terlihat laporan respon waktu dari masing-masing *script*. *Response vs. time report* dapat dilihat pada gambar 5-35.



Gambar 5-35. Response vs time report dari performance testing 100 virtual user

Pada pengujian selanjutnya ada beberapa hal yang ditemui:

- Pengujian pada *Local computer* tidak dapat dilakukan dengan *virtual user* lebih dari 255 karena keterbatasan memori.
- Pada pengujian pada *Local computer* dengan jumlah virtual user 200, hasilnya adalah komputer *error* tidak bergeming.
- Kemampuan pengujian tergantung juga pada spesifikasi komputer pengujinya.

BAB VI ANALISA PENGUJIAN

Pada bab ini, akan dipaparkan hasil analisa penggunaan *software* Rational berdasarkan percobaan yang telah dilakukan. Akan dibahas juga perbandingannya dengan pengujian manual dan tesis yang ada sebelumnya.

6.1 Analisa Penggunaan Rational

Software Rational dikembangkan dengan tujuan untuk membantu para pengembang *software* dalam melakukan pengujian terhadap *software* yang mereka buat. Dengan kualitas pengujian yang baik, maka kualitas *software* yang dibuat pun akan semakin baik. Dengan menggunakan *software* penguji seperti Rational ada beberapa keuntungan yang akan didapat, yaitu:

- Mempercepat waktu pengujian. Dengan menggunakan Rational, penguji *software* tidak perlu repot untuk mempelajari setiap kode yang ada pada *software* yang akan diuji. Dengan memahami use case dan menerapkannya pada TestManager, penguji dapat dengan mudah dan cepat melakukan pengujian.
- Efektifitas tinggi dalam melakukan pengujian. Dalam pengujian dengan cara manual, seringkali pengujian yang dilakukan kurang efektif karena keterbatasan pengujian. Dengan menggunakan Rational akan lebih efektif.
- Efektifitas sumber daya pengujian. Pengujian dengan cara manual memakan biaya yang besar jika benar-benar dilakukan. Pengembang harus mempekerjakan orang khusus untuk mempelajari dan melakukan pengujian. Seringkali waktu pembelajaran juga menjadi lama karena keterbatasan penguji. Penggunaan Rational akan memangkas sumber daya yang dibutuhkan. Waktu untuk mempelajari penggunaan Rational relatif lebih cepat. Apalagi penguji yang sudah memiliki kemampuan dalam menggunakan Rational hanya cukup mempelajari use case dan perilaku *software*.

Penggunaan Rational untuk menguji aplikasi web akan sangat efektif. Hanya dengan memasukkan URL dari aplikasi yang akan diuji dan membuat *test*

case untuk pengujian, penguji sudah dapat menguji aplikasi yang akan diuji. Rational menyediakan dukungan yang baik terhadap aplikasi web.

6.2 Perbandingan antara penelitian penulis dengan penelitian sebelumnya

Sebelum penelitian dilakukan, telah ada penelitian-penelitian sebelumnya yang menggunakan Rational TestManager dan Robot. Penelitian yang telah menggunakan Rational:

- Pengujian Kinerja Digital Library Online dan Kesatuan Jaringan dan Basis Data Menggunakan Alat Bantu Rational Robot dan Rational TestManager oleh Robertus Nursukma [NUR05]. Ruang lingkup dari tesis ini adalah pada pengujian kinerja LONTAR. Karena keterbatasan waktu, pengujian yang dilakukan hanya dilakukan pada dua fungsi, yaitu *most frequency usage* dan *complexity of function* untuk menggambarkan 80% karakteristik kinerja LONTAR berdasarkan 20% fungsi yang paling sering digunakan.

Modul yang akan diuji dipilih berdasarkan pooling dari admin LONTAR. Karena beberapa kendala, hanya modul Pencarian koleksi biasa yang diuji. Kesimpulan dari pengujian adalah bahwa hasil kinerja dengan aplikasi yang bersifat *read-only* atau hanya berhubungan dengan pencarian telah menunjukkan bahwa kinerja LONTAR dan jaringan serta basis data tidak mengalami penurunan drastis dan tergolong normal. Penurunan kinerja pada pengujian karena faktor lain antara lain kesalahan desain arsitektur perangkat keras dan lunak.

- *Software Testing* Berdasarkan Metodologi Rational Unified Process Menggunakan Alat Bantu Rational TestManager dan Rational Robot oleh Adi Pratomo [PRA06].

Latar belakang penelitian karena kendala-kendala yang ada pada *software testing*. Ruang lingkup penelitian adalah:

- Menggunakan RUP dan *workflow* test dalam RUP termasuk Rational TestManager dan Robot.
- Aplikasi yang diuji adalah Webmail UI.

- Hasil akhir adalah tutorial mengenai *software testing* berdasarkan RUP dengan menggunakan Rational TestManager dan Robot.
- Tdak membahas mengenai hasil akhir.
- Pengerjaan hanya sebatas pembuatan tutorial proses.

Tujuan dari penelitian adalah untuk menyusun panduan atau tutorial (*Computer Aided learning* atau *Computer Based Learning*) dalam melakukan perencanaan, persiapan dan pengujian berdasarkan *workflow testing* pada RUP. *Test case* yang dipilih dalam melakukan pengujian tidak berdasarkan apapun. Hanya pilihan penulis saja.

Pada tesis sebelumnya dalam menentukan *test case* yang akan diuji tidak berdasarkan suatu cara yang metodologi yang baku [GAL04]. Pada pengujian Lontar *test case* yang dipilih berdasarkan hasil poling dan pada Webmail UI *test case* dipilih sesuai pilihan penelitiannya. Sedangkan pada penelitian yang penulis lakukan, *test case* yang diuji berdasarkan konsep *full line coverage*. Dengan konsep tersebut, penulis dapat menguji seluruh fungsi yang ada hanya dengan jumlah *test case* yang minimum.

Pengujian pada tesis sebelumnya juga tidak menggunakan komputer *agent*. Pengujian hanya dilakukan pada *local computer*. Local computer yang digunakan tentunya mempunyai batas maksimum virtual tester yang dapat digunakan, sehingga pengujian hanya dapat dilakukan sebatas kemampuan computer tersebut saja. Pada pengujian LONTAR dan Webmail hal tersebut belum dapat menunjukkan performa yang menyeluruh, karena pada kenyataannya kedua aplikasi tersebut dapat diakses dan digunakan oleh lebih dari satu komputer. Dengan menggunakan banyak komputer *agent*, jumlah virtual tester untuk melakukan pengujian dapat lebih diperbanyak, sehingga pengujian akan lebih maksimal.

Dari ketiga aplikasi yang telah diuji sebelumnya pada tesis yang ada sebelumnya, yaitu LONTAR, Webmail UI dan SIMBION, Rational akan paling efektif digunakan pada SIMIBION. Hal tersebut dikarenakan *output* dari SIMBION lebih bersifat statis. Pada LONTAR, hasil pencarian memiliki probabilitas yang berbeda-beda. Pada aplikasi Webmail UI, pengujian hanya

dilakukan pada sisi *client* saja. Padahal, fungsi utama dari Webmail adah untuk mengirim *email*. Pengujian menggunakan Rational pada sisi *client* tidak dapat memastikan bahwa *email* telah terkirim. Pengujian pada Webmail harus dilakukan pada *server* dan *client* untuk dapat menguji seluruh fungsi yang ada.

