
MEMBANGUN SOFTWARE APLIKASI PADA ANTRIAN JARINGAN JACKSON UNTUK MENENTUKAN PERFORMANSI OPTIMAL

***Gumgum Darmawan,** Okira Mapanta, ***Trifandi Lasalewo**

* Universitas Padjadjaran, gumstat@yahoo.com, ** Universitas Widyatama, daoki_02@yahoo.co.id *** Universitas Gorontalo,

ABSTRAK

Dalam tugas akhir ini dibangun sebuah *software* aplikasi sistem antrian, untuk memudahkan dan mempercepat dalam melakukan proses perhitungan performansi pada antrian jaringan dengan metode Jackson. Aplikasi ini digunakan untuk menganalisis performansi sistem antrian yang dapat menghasilkan informasi dari ukuran-ukuran performansi antrian. Dengan adanya aplikasi ini akan dapat membantu peneliti atau pihak yang berkepentingan dengan persoalan antrian.

Paradigma pembangunan *software* aplikasi adalah *waterfall paradigm*, sedangkan untuk kegiatan desain sistem, proses pemodelannya menggunakan *Data Flow Diagram* (DFD). Adapun *tool* aplikasi yang digunakan untuk implementasi sistem antrian ini menggunakan program microsoft visual basic 6.0.

Kata kunci : Antrian Jaringan Jackson, Data Flow Diagram (DFD), Microsoft Visual Basic

1. PENDAHULUAN

Antrian jaringan merupakan sekelompok workstation dimana pelanggan/pendatang dapat berpindah dari satu workstation ke workstation lebih dari satu kali. Workstation merupakan sarana pelayanan yang berada pada sistem antrian jaringan dimana pada sistem antrian jaringan terdapat lebih dari satu workstation.

Antrian jaringan (*Queueing Network*) telah banyak dikaji oleh para peneliti seperti Jackson, J.R. (1957), mengkaji karakteristik dari antrian jaringan, Kelly (1975) yang mengkaji karakteristik konsumen/pendatang pada antrian jaringan. Lemoine (1977) yang mengkaji keseimbangan pada suatu antrian jaringan, Perros (1994) yang mengkaji *blocking system* pada sistem antrian jaringan.

Salah satu jenis antrian jaringan yang menarik dikaji adalah Antrian Jaringan Jackson dimana setiap workstation mempunyai pelayanan tunggal dengan konsumen dapat berpindah dari workstation satu ke workstation lainnya dapat lebih dari satu

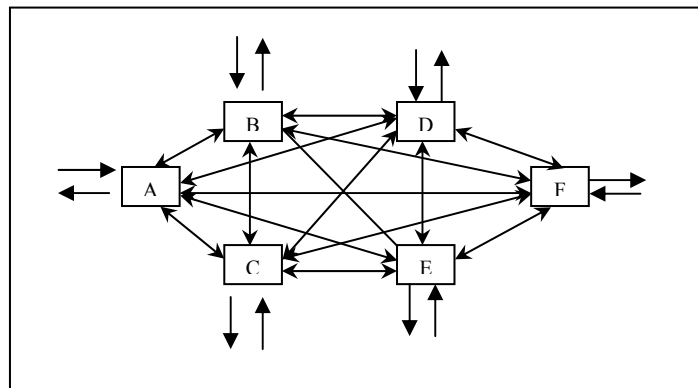
kali. Antrian Jaringan Jackson berdasarkan sumber kedatangan konsumen terbagi menjadi dua yaitu Antrian Jaringan Jackson terbuka (*Open Jackson Networks*) dan Antrian Jaringan Jackson tertutup (*Closed Jackson Networks*). Antrian Jaringan Jackson terbuka (*Open Jackson Networks*) pendaang/konsumen berdatangan dari luar dan dalam sistem itu sendiri, sedangkan Antrian Jaringan Jackson tertutup (*Closed Jackson Networks*), konsumen/pendaang berpindah dari workstation ke workstation lainya hanya didalam sistem itu sendiri.

Antrian Jaringan Jackson terbuka (*Open Jackson Networks*) telah banyak dikaji seperti Burke (1969), mengkaji tiga workstation dengan workstation pertama dan ketiga mempunyai pelayanan tunggal dan pelayanan kedua mempunyai pelayanan multipel, Simon dan Foley (1979), yang mengkaji tiga workstation dengan pelayanan tunggal. Antrian Jaringan Jackson tertutup (*Closed Jackson Networks*) telah dikaji oleh Buzen (1973) dan Bruell dan Balbo (1980) yang membuat algoritma komputasi dari Antrian Jaringan Jackson tertutup (*Closed Jackson Networks*).

Pada penelitian ini akan dikaji Antrian Jaringan Jackson terbuka (*Open Jackson Networks*) dengan multi server yang mengacu pada Kulkarni (1999). Sistem terdiri atas enam (6) buah workstation dengan pelayanan lebih dari satu.

2. ANTRIAN JARINGAN JACKSON

Antrian Jaringan adalah sebuah antrian dimana konsumen dapat pindah dari satu workstation ke workstation lain beberapa kali sebelum meninggalkan sistem. Pada antrian ini terdapat lebih dari satu workstation.



Gambar 1. Skema Antrian Jaringan Jackson dengan Enam Workstation

Asumsi pada Antrian Jaringan Jackson

- ✓ Jaringan mempunyai N pelayanan tunggal
- ✓ Stasiun ke-i mempunyai pelayan sebanyak s_i .
- ✓ Setiap stasiun mempunyai ruang tunggu tak terbatas.
- ✓ Pelanggan datang pada stasiun ke-i dari luar sistem dengan tingkat kedatangan $P(\lambda_i)$ dengan semua kedatangan bersifat independent.
- ✓ Waktu pelayanan pada stasiun ke-i berdistribusi iid $Exp(\mu_i)$.
- ✓ Konsumen keluar dari workstation ke-i dan sampai ke workstation ke-j dengan peluang $p_{i,j}$ yang bersifat bebas untuk setiap workstation.

Langkah-langkah penentuan Performansi Antrian Jaringan Jackson

2.1 Menentukan Tingkat kedatangan

$$a_i = \lambda_i + b_i, \quad 1 \leq i \leq N$$

$$\lambda_i = \lambda_{tot} \gamma_i / \sum_{i=1}^N \gamma_i$$

Nilai-nilai parameter pada sistem meliputi,

a_i = tingkat kedatangan total pada workstation ke-i,

s_i = Banyaknya fasilitas pelayanan workstation ke-i,

λ_i = Tingkat kedatangan eksternal pada workstation ke-i,

b_i = Tingkat kedatangan internal pada workstation ke-i,

γ_i = *Arrangement Code* (1 jika terbuka, 0 jika tertutup),

N = Banyaknya workstation,

λ_{tot} = Tingkat kedatangan eksternal total pada sistem.

$$b_j = \sum_{i=1}^N a_i p_{i,j}, \quad 1 \leq j \leq N$$

$$a_j = \lambda_j + \sum_{i=1}^N a_i p_{i,j}, \quad 1 \leq j \leq N$$

dengan $a = [a_1, a_2, \dots, a_N]$

$$\lambda = [\lambda_1, \lambda_2, \dots, \lambda_N]$$

Sehingga,

$$a = \lambda + aP$$

$$a(I - P) = \lambda$$

$$a = \lambda(I - P)^{-1}$$

2.2 Menentukan Matriks Transisi Jackson

Matriks Transisi Jackson menunjukkan besarnya peluang perpindahan didalam sistem antrian, mempunyai bentuk sebagai berikut,

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} & \dots & p_{1,N} \\ p_{2,1} & p_{2,2} & p_{2,3} & \dots & p_{2,N} \\ p_{3,1} & p_{3,2} & p_{3,3} & \dots & p_{3,N} \\ \dots & \dots & \dots & \dots & \dots \\ p_{N,1} & p_{N,2} & p_{N,3} & \dots & p_{N,N} \end{bmatrix}, \sum_{j=1}^N p_{i,j} = 1, 1 \leq i \leq N.$$

2.3 Menentukan Stabilitas Sistem

Antrian Jaringan Jackson dikatakan stabil jika, matriks **I-P** invertibel dengan P adalah matriks transisi Jackson network dan $a_i < s_i \mu_i$ untuk semua $i = 1, 2, \dots, N$ dengan $a = [a_1, a_2, \dots, a_N]$. Dengan kata lain Jackson Network disebut stabil jika $\frac{a_i}{\mu_i} < s_i$, untuk $i = 1, 2, \dots, N$.

2.4 Menentukan Ukuran Performansi Sistem antrian.

Ukuran performansi antrian merupakan ukuran yang menunjukkan efektifitas dan efisiensi dari antrian. Ukuran performansi antrian untuk model (M/M/s):(FCFS/~/~) adalah,

$$a_n = a, \quad n = 0, 1, 2, \dots$$

$$\mu_n = \begin{cases} n\mu & 0 \leq n \leq s \\ s\mu & n \geq s \end{cases}$$

Jika $a < s\mu$, maka hasil steady state-nya adalah

$$P_0 = \frac{1}{\sum_{n=0}^{s-1} \frac{(a/\mu)^n}{n!} + \frac{(a/\mu)^s}{s!} \sum_{n=s}^{\infty} \left(\frac{a}{s\mu}\right)^{n-s}}$$

$$= \frac{1}{\sum_{n=0}^{s-1} \frac{(a/\mu)^n}{n!} + \frac{(a/\mu)^s}{s!} \frac{1}{1 - a/s\mu}}$$

$$P_n = \begin{cases} \frac{(a/\mu)^n}{n!} P_0 & \text{jika } 0 \leq n < s \\ \frac{(a/\mu)^n}{s! s^{n-s}} P_0 & \text{jika } n \geq s \end{cases}$$

Dengan $\rho = a/s\mu$, maka

$$L_q = \frac{\left(\frac{a}{\mu}\right)^s \rho}{s!(1-\rho)^2} P_0, W_q = \frac{L_q}{a}, W = W_q + \frac{1}{\mu}$$

$$L = a\left(W_q + \frac{1}{\mu}\right)$$

$$= L_q + \frac{a}{\mu}$$

Dengan

P_0 = Peluang tidak terdapat konsumen/pendatang pada sistem antrian,

P_n = Peluang terdapat ada n konsumen pada sistem antrian,

L_q = Rata-rata banyaknya konsumen yang mengantri pada sistem antrian,

L_s = Rata-rata banyaknya konsumen yang mengantri ditambah dengan konsumen yang sedang dilayani pada sistem antrian,

W_q = Rata-rata lamanya konsumen menunggu sampai dilayani,

W = Rata-rata lamanya konsumen menunggu dan dilayani,

ρ = Utilitas Sistem (tingkat kesibukan pelayanan).

2.5 Menentukan Pelayanan yang mengganggu

Untuk menentukan banyaknya pelayanan yang mengganggu dapat digunakan persamaan sebagai berikut;

$$Idle_i = s_i - (L_s(i) - L_q(i)), \text{ dengan } 1 \leq i \leq N.$$

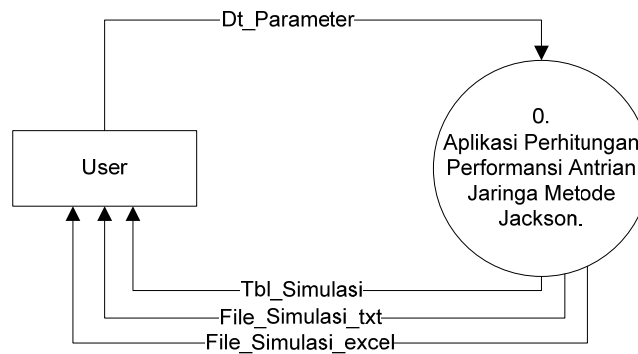
$Idle_i$ adalah banyaknya pelayanan yang mengganggu pada workstation ke-i.

3. Diagram Data

Data diagram adalah suatu bagan yang menggambarkan arus data dalam suatu sistem, yang digambarkan dengan sejumlah simbol tertentu untuk menunjukkan perpindahan data yang terjadi dalam proses suatu sistem bisnis. Perancangan proses digambarkan dengan menggunakan *Data Context Diagram* (DCD) dan *Data Flow Diagram* (DFD).

3.1 Data Context Diagram (DCD)

DCD adalah diagram yang menggambarkan proses secara keseluruhan dari suatu *software* bantu dan hubungannya dengan lingkungan. Pada gambar berikut adalah DCD aplikasi perhitungan performansi antrian jaringan model Jackson yang akan dibangun:

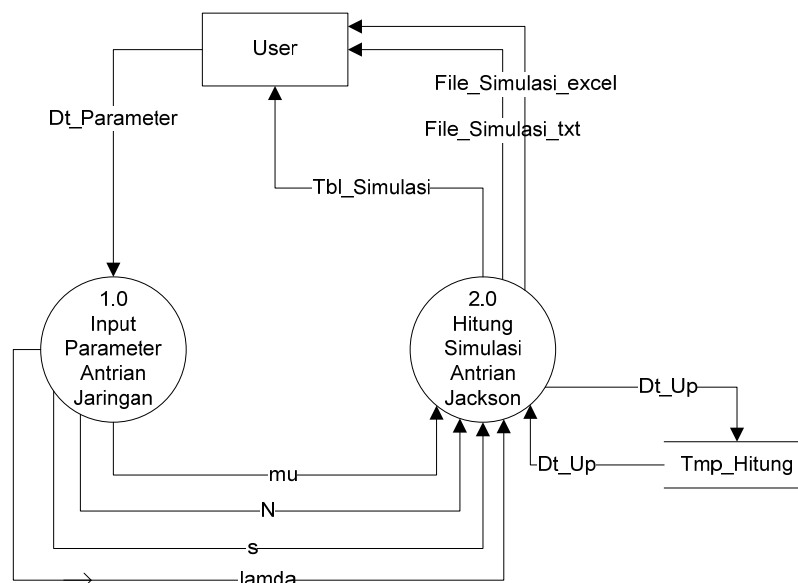


Gambar 2. Data Context Diagram (DCD)

Pada DCD ini dijelaskan *user* adalah sebagai eksternal *entity* dari sistem aplikasi perhitungan performansi antrian jaringan metode Jackson ini. *User* melakukan proses input data parameter antrian kemudian diproses didalam sistem aplikasi perhitungan performansi antrian jaringan metode Jackson. Dan *user* menerima informasi berupa tabel informasi dan dapat menyimpan kedalam bentuk *file text* dan *file excel*.

3.2 Data Flow Diagram (DFD)

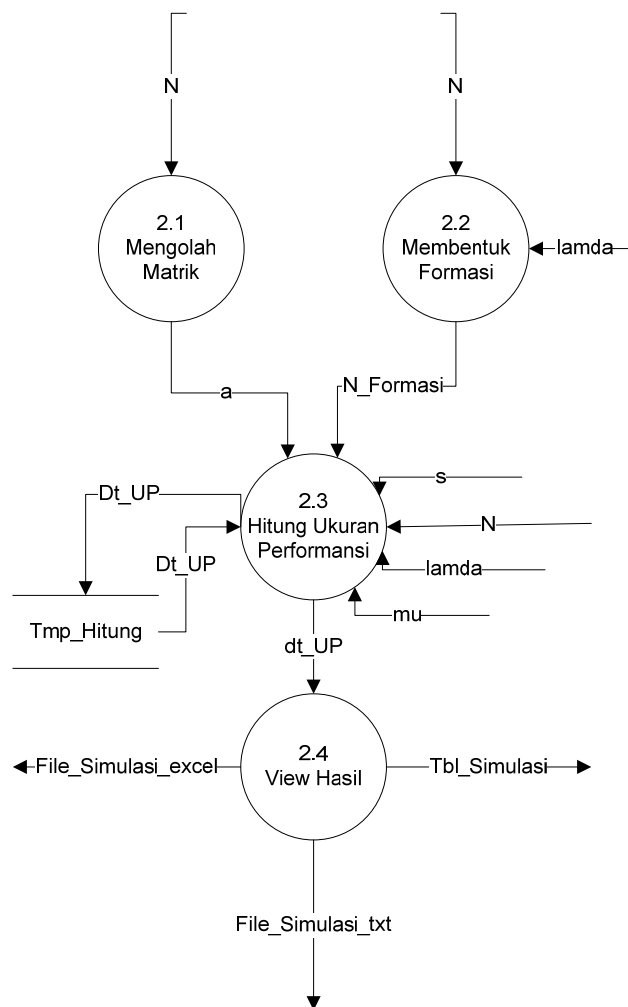
DFD adalah diagram yang akan menggambarkan aliran data antara proses dan lingkungannya. Pada gambar dibawah ini adalah DFD dari aplikasi perhitungan antrian jaringan metode Jackson.



Gambar 3. Data Flow Diagram (DFD) Level 1

Pada DFD level 1 ini proses yang ada di DCD dipecah menjadi beberapa proses yaitu proses 1.0 Input Parameter Antrian dan proses 2.0 Hitung simulasi antrian Jackson. Sebagai *eksternal entity* user melakukan proses input Dt_parameter kedalam proses 1.0 *Input* Parameter Antrian. Pada proses 1.0 sistem mengirimkan inputan parameter antrian dari *user* yaitu terdiri dari μ , λ , N , s yang dikirimkan kedalam proses 2.0 Hitung Simulasi Antrian. Setelah dikirimkan kedalam proses 2.0 kemudian sistem melakukan perhitungan dan akan ditampilkan ke *user* berupa *tbl_simulasi*, *file_simulasi_txt*, dan *file_simulasi_excel*.

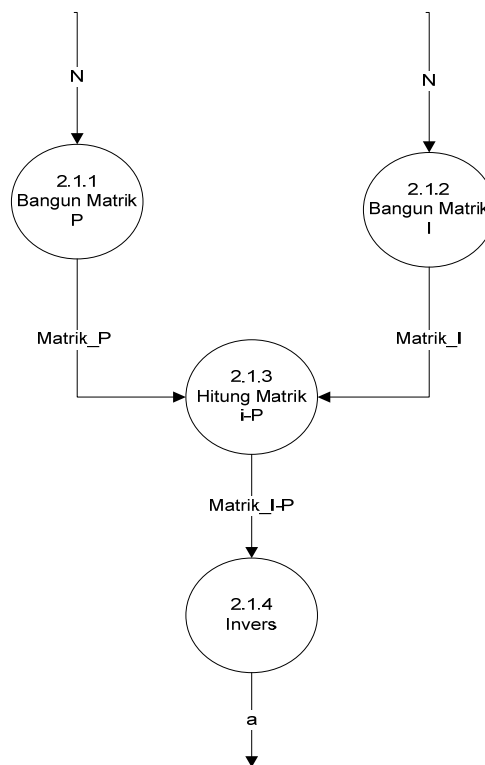
3.3 Data Flow Diagram (DFD) level 2 proses 2



Gambar 4. DFD Level 2 Proses 2

Pada DFD level 2 ini dijelaskan proses 2.0 dipecah menjadi beberapa proses yaitu proses 2.1 Mengolah Matrik, proses 2.2. Membentuk formasi, proses 2.3. Hitung Ukuran Performansi, proses 2.4. *View* Hasil. Pada proses 2.1. Proses ini menerima inputan dari *user* dan diolah dari proses sebelumnya yaitu inputan *N* kemudian pada proses ini dilakukan proses pengolahan matrik sesuai jumlah inputan *dt_parameter* (*N*) dan mengeluarkan *output* *a*. pada proses 2.2. menerima inputan juga berupa *N* dan akan diolah untuk melakukan proses membentuk formasi atau pengaturan kedatangan dengan *output* berupa *N_formasi* (1 = terbuka ,0 = tertutup). Pada proses 2.3. ini menerima inputan dari proses sebelumnya yaitu berupa data *a*, dan *N_formasi*. Proses 2.3. akan dilakukan perhitungan ukuran performansi antrian dan mengeluarkan *output* berupa data *Dt_Up* (data ukuran performansi). *Output* disimpan sementara kedalam *tmp_hitung* dan dikirimkan ke proses berikutnya yaitu proses 2.4. *View* hasil. Pada proses 2.4. kita dapat mendapatkan informasi dalam bentuk *tbl_simulasi*, dan menyimpan kedalam *file_simulasi_txt*, *file_simulasi_excel*.

3.4 Data Flow Diagram (DFD) level 3 proses 2.1

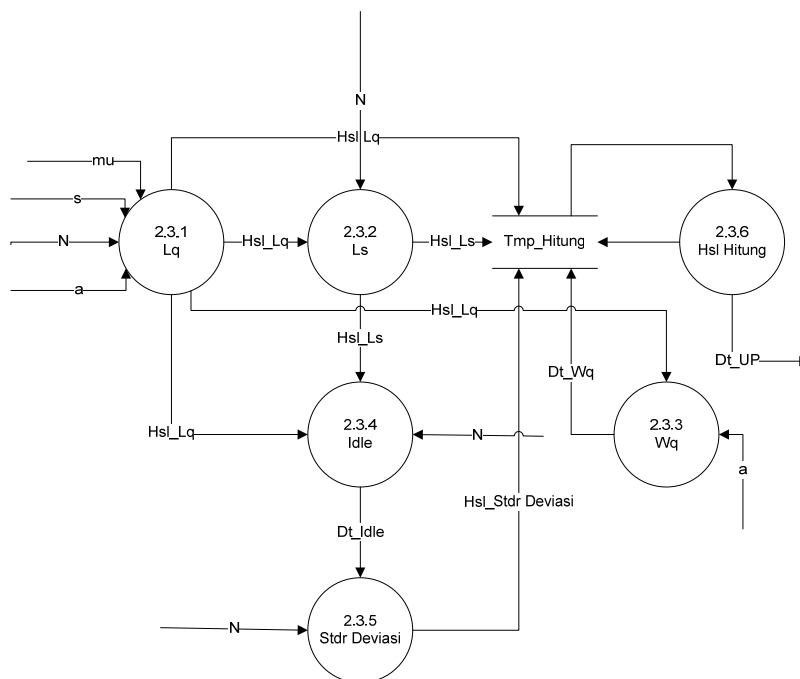


Gambar 5. DFD Level 3 Proses 2.1

Pada DFD level 3 Proses 2.1. Mengolah Matrik yang dirinci menjadi beberapa proses turunan yaitu terdiri dari proses 2.1.1 Bangun Matrik P, proses 2.1.2 Bangun Matrik I, proses 2.1.3 Hitung Matrik I-P dan proses 2.1.4 *Invers* Matrik. Pada proses 2.1.1 menerima inputan yaitu N sebagai olahan membangun matrik P dan akan mengeluarkan *output* berupa Matrik P. Proses 2.1.2 Menerima inputan N juga diolah untuk membangun matrik I dan mengeluarkan *output* Matrik I. Proses 2.1.3 Menghitung Matrik I-P menerima *input* dari proses 2.1.1 dan proses 2.1.2. dan mengeluarkan *output* Matrik hasil hitung I-P. Setelah proses ini akan dikirimkan ke proses selanjutnya untuk dilakukan perhitungan *invers* Matrik pada proses 2.1.4. yang akan mengeluarkan *output* a (hasil perhitungan *invers*).

3.5 Data Flow Diagram (DFD) level 3 proses 2.3

Pada DFD level 3 proses 2.3. Hitung Ukuran Performansi dijelaskan proses ini menurunkan beberapa proses yaitu proses 2.3.1 Hitung Lq, proses 2.3.2. Hitung Ls, proses 2.3.3. Hitung Wq, proses 2.3.4. Hitung *idle*, proses 2.3.5. Hitung Stdr Deviasi. Setelah proses tersebut dijalankan maka akan proses hitung tersebut mengirimkan data kedalam Tmp_hitung dan dari Tmp_hitung dikirimkan kembali ke proses Hsl_hitung kemudian diproses dan mengeluarkan *output* Dt_Up.



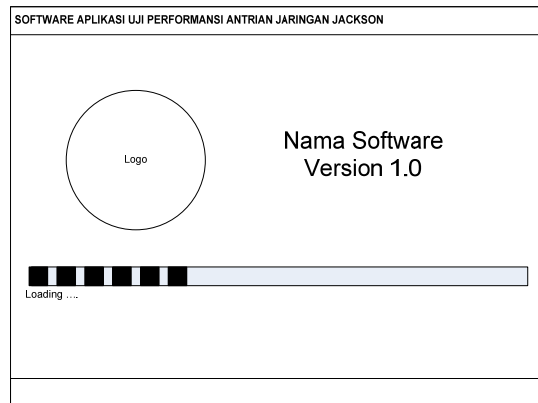
Gambar 6. DFD Level 3 Proses 2.3

4 Perancangan Antar Muka

Perancangan antarmuka digunakan untuk menjelaskan rutinitas program yang akan dijalankan oleh sebuah sistem dan untuk menjelaskan interaksi antar *user* dengan aplikasi.

4.1 Desain Antarmuka Awal

Perancangan antarmuka awal ini adalah untuk tampilan di *window* awal ketika sistem dijalankan. Pada antarmuka ini ditampilkan judul dari aplikasi dan menggubakan proses waktu me
nunggu sebelum memasuki sistem.



Gambar 7 Antarmuka Awal

4.2 Desain Antarmuka *Form Input*

Gambar 8. Antarmuka *Form Input*

Antarmuka ini merupakan *form* antarmuka *user* dengan sistem dalam melakukan proses inputan parameter antrian sebelum dilakukan proses hitung simulasi. Antarmuka ini merancang bentuk tampilan *form* yang terdiri dari inputan jumlah *worksatation* yang terdiri dari maksimal 6. Kemudian inputan tingkat kedatangan per jam berupa *input* nilai parameter. Inputan fasilitas pelayanan sesuai jumlah *worksatation* yang diinputkan. Dan waktu pelayan juga dirancang inputan nilai parameter sesuai jumlah inputan *workstation*. Dan beberapa tombol yaitu tombol acak, tombol simulasi, tombol tabel dan tombol keluar.

4.3 Desain Antarmuka Tabel Hasil

Tabel Ukuran Performansi Antrian Jaringan Jackson				
Blocking System	Total Idle	Rata-rata Idle	Standar Deviasi	Lq
TABEL UKURAN PERFORMANSI				
<input type="button" value="Simpan"/> <input type="button" value="Cetak"/> <input type="button" value="Tutup"/>				

Gambar 9. Antarmuka Tabel Hasil

Pada antarmuka tabel hasil ini akan ditampilkan tabel hasil perhitungan setelah *user* melakukan proses simulasi. Tabel ini juga dapat ditampilkan melalui tampilan dari tombol rancangan sebelumnya yaitu tombol tabel hasil. Pada rancangan antarmuka akan dibuat tombol simpan untuk menyimpan hasil tabel, tombol cetak untuk mencetak hasil tabel dan tombol tutup untuk keluar dari *form* antarmuka tabel hasil ini.

5 HASIL

Setelah Software Aplikasi dibangun didapat beberapa nilai output untuk Jaringan Jackson dengan enam (6) workstation sebagai berikut;

Tabel 1. Ukuran Performansi Dari Antrian Jaringan Jackson

γ (N=6)	Idle	Rata-rata	Deviasi Standar	L_q	L_s	W_q
(1,1,1,1,1,1)	28,30	4,72	2,97	2,82	27,26	0,005
(1,1,1,0,0,0)	30,70	5,12	6,15	10,04	34,09	0,015
(0,1,1,1,0,0)	22,10	3,69	2,22	9,64	35,12	0,014
(1,1,0,1,0,0)	20,70	3,45	2,53	11,70	37,41	0,017
(1,1,0,0,0,1)	30,20	5,04	6,40	12,16	36,29	0,018
(1,0,1,1,0,0)	25,00	4,16	2,90	5,26	30,26	0,008
(0,1,1,0,0,1)	31,60	5,28	6,22	10,10	33,99	0,015
(0,0,1,1,0,1)	25,90	4,32	3,10	5,28	30,16	0,008
(1,1,1,1,1,0)	27,00	4,50	2,64	4,93	29,60	0,008
(1,1,1,1,0,1)	26,71	4,45	2,11	1,73	26,44	0,003
(1,1,1,0,1,1)	32,71	5,45	5,89	5,12	28,84	0,008
(1,1,0,1,1,1)	26,71	4,45	2,93	5,19	29,91	0,008
(1,0,1,1,1,1)	29,28	4,88	3,40	4,86	29,15	0,008
(1,0,1,1,1,1)	27,57	4,59	2,84	4,96	29,53	0,008
(1,1,1,1,0,0)	24,64	4,11	1,60	2,50	27,56	0,004
(0,1,1,1,0,1)	25,36	4,23	1,96	2,54	27,48	0,004
(1,0,1,1,0,1)	27,50	4,58	2,76	2,24	26,82	0,003
(1,1,0,1,0,1)	24,29	4,04	2,16	3,02	28,14	0,005

Berdasarkan hasil perhitungan dengan menggunakan Visual Basic (Untuk Enam workstation). Jika antrian dititik beratkan pada pengurangan fasilitas yang mengganggu maka lebih baik melakukan pengaturan dengan $\gamma = (1,1,0,1,0,0)$. Jika antrian dititik beratkan pada cepatnya menunggu untuk mengantri maka lebih baik melakukan pengaturan dengan $\gamma = (1,1,1,1,0,1)$, artinya hanya pintu ke lima yang ditutup sedangkan yang lainnya di buka.

DAFTAR PUSTAKA

Bruell SC & Balbo G. 1980. *Computational Algorithm for Closed Queueing Networks. Operating and Programming System Series*. P.J.Denning (Ed.). New York. Oxford:North Holland.

Burke PJ. 1969. The Dependence of Service in Tandem M/M/s Queues. *Operational Research*.**17**:754-755.

Buzen JP. 1973. Computational Algorithms for Closed Queueing Networks with Exponential Servers. *Communication*. ACM **16** : 527-531.

Jackson JR. 1957. Networks of Waiting Lines. *Operational Research*.**5** : 518-521.

Kulkarni VG. 1999. *Modeling, Analysis, Design, and Control of Stochastic System*. Springer-Verlag New York USA.

Kelly FP. 1975. Networks of Queues with Customers of Different Types. *Journal of Applied Probability*.**12** : 542-554.

Lemoine AJ.1977. Networks of Queues-A Survey of Equilibrium Analysis. *Management Science*.**24** : 464-481.

Perros H. 1994. *Queueing Networks with Blocking*. New York:Oxford University Press.

Simon B & Foley RD. 1979. Some Results on Sojourn Times in Cyclic Jackson Networks. *Management Science*. **25** : 1027-1034.

Lampiran 1.

Kamus Data

No	Data	Kamus Data
1	N	N = [Pilihan jumlah workstation] N(int) = [2 3 4 5 6] * Jumlah workstation (N) max 6
2	s	s = banyaknya fasilitas pd setiap workstation Tipe Data : s (int) *Nilai Acak Parameter Antrian s
3	mu (μ)	mu = waktu pelayanan pd setiap workstation mu (int) *Nilai Acak Parameter Antrian (μ)
4	Lamda (λ)	Lamda = Total kedatangan Lamda (int) *Nilai Parameter Antrian lamda (λ)
5	Tbl_Simulasi	*Tabel Ukuran Performansi
6	File_simulasi_txt	*Hasil Simpan type *.txt
7	File_simulasi_excl	*Hasil Simpan type *.excl
8	Dt_Parameter	Dt_Parameter = [N+s+mu+lamda] *Parameter Inputan Antrian
10	a	Matrik a = a(i,j) *Hasil perhitungan invers Matrik
11	N_formasi	N_Formasi = code [1+0] *Arrangement code untuk menentukan pengaturan kedatangan code 1 = terbuka, code 0 = tertutup.
12	Dt_UP	*Hasil Perhitungan simulasi
13	Matrik I	Matrik I = I(i,j) *Matrik I
14	Matrik P	Matrik P = P (i,j) *Matrik P
15	Matrik I-P	Matrik I-P = I(i,j) – P (i,j) *Matrik I-P
16	Matrik a	*Matrik a Hasil invers MatrikI-P
17	Hsl_Lq	*Hasil perhitungan Lq1+Lq2... LqN (Rata-rata banyaknya konsumen yg mengantri)
18	Hsl_Ls	*Hasil perhitungan Ls1+Ls2... LsN (Rata-rata banyaknya konsumen yg mengantri + dilayani)
19	Dt_Wq	*Hasil perhitungan Wq1+Wq2... WqN (Rata-rata lamanya konsumen menunggu)
20	Dt_idle	*Hasil perhitungan idle (Menentukan pelayanan yang menganggur)

21	Hsl_stdDeviasi	*Hasil perhitungan standar deviasi
22	Tmp_Hitung	*Penyimpanan hasil perhitungan secara temp (sementara)

Lampiran 2

Proses spesifikasi (PSPEC)

Tabel 2.1 PSPEC Level 1 Proses 1.0 Input Parameter Antrian Jaringan

Aliran	Data
IN	Dt_Parameter
OUT	N,s,mu,lamda
Proses	
Input Pilihan Jumlah N Input s Input mu (μ) Input lamda(λ) If Pilihan Jumlah workstation N:[2 3 4 5 6] = true then Tombol Acak = Aktif input s >= N input mu >= N input lamda Else Pesan "Jumlah workstation belum dipilih" End if Output Dt_parameter	

Tabel 2.2. PSPEC Level 1 Proses 2.0 Hitung Simulasi Antrian Jackson

Aliran	Data
IN	N,s,lamda,mu
OUT	File_simulasi_txt, File_simulasi_excl, Tbl_simulasi
Proses	
Input N,s,lamda,mu Hitung simulasi Antrian Jackson Lakukan Proses olah Matrik Lakukan Proses Bentuk formasi Lakukan Proses Hitung Lq,Ls,Wq,idle,Std Deviasi Output file_simulasi_txt, File_simulasi_excl,Tbl_Simulasi	

Tabel 2.3. PSPEC Level 2 Proses 2.1 Mengolah Matrik

Aliran	Data
IN	N
OUT	Matrik_a

Proses
Input N Bangun Matrik P Bangun Matrik I Hitung Matrik I-P Hitung Invers I-P Output Matrik_a

Tabel 2.4. PSPEC Level 2 Proses 2.2 Membentuk Formasi

Aliran	Data
IN	N
OUT	N_formasi
Proses	
Input N N_formasi = 2^N 'Arrangement code (1 = pintu terbuka, 0 = pintu tertutup) Output N_formasi	

Tabel 2.5. PSPEC Level 2 Proses 2.3 Hitung Ukuran Performansi

Aliran	Data
IN	a, N_formasi, Dt_Parameter
OUT	dt_UP
Proses	
Input a, Input N_formasi, Input Dt_Parameter Hitung Lq, Ls, Wq, idle, Stdr Deviasi Output dt_UP	

Tabel 2.6. PSPEC Level 2 Proses 2.4 View Hasil

Aliran	Data
IN	Dt_UP
OUT	File_simulasi_txt, file_simulasi_excl, Tbl_simulasi
Proses	
Input Dt_UP Proses View Output file_simulasi_txt Output file_simulasi_excl Output Tbl_simulasi	

Tabel 2.7. PSPEC Level 3 Proses 2.1.1 Bangun Matrik P

Aliran	Data
IN	N
OUT	Matrik_P
Proses	
Input N Matrik_P = 1/N For i 1 to N For j 1 to N If i=j then Matrik_P= 0 Else Matrik_P = 1/N Next j Next i Output Matrik_P	

Tabel 2.8. PSPEC Level 3 Proses 2.1.2 Bangun Matrik I

Aliran	Data
IN	N
OUT	Matrik_I
Proses	
Input N Matrik_I = 1 For i 1 to N For j 1 to N If i=j then Matrik_I= 1 Else Matrik_I = 0 Next j Next i Output Matrik_I	

Tabel 2.9. PSPEC Level 3 Proses 2.1.3 Hitung Matrik I-P

Aliran	Data
IN	Matrik_I, Matrik_P
OUT	Matrik_I-P
Proses	
Input Matrik_I Input Matrik_P For i 1 to N For j 1 to N	

<p>Hitung I-P = Matrik_I(i,j)-Matrik_P(l,j) Next j Next i Output Matrik I-P</p>
--

Tabel 2.10. PSPEC Level 3 Proses 2.1.4 Invers

Aliran	Data
IN	Matrik_I-P
OUT	A
Proses	
Input Matrik_I-P Proses Invers 'Gauss Elimination method Output a	

Tabel 2.11. PSPEC Level 3 Proses 2.3.1 Lq

Aliran	Data
IN	N,s,mu,a
OUT	Hsl_Lq
Proses	
Input N,s,mu,lamda Hitung $Lq = ((a/\mu)^s * (a/s * \mu)) / (s_fact(1-a/s * \mu)) * P0$ $Lq1 = Lq1 + Lq$ Output Hsl_Lq	

Tabel 2.12. PSPEC Level 3 Proses 2.3.2 Ls

Aliran	Data
IN	Hsl_Lq, N,a,mu
OUT	Hsl_Ls
Proses	
Input Hsl_Lq,N,a,mu Hitung $Ls = a * ((Lq / a) + (1/\mu))$ $Ls1 = Ls1 + Ls$ Output Hsl_Ls	

Tabel 2.13. PSPEC Level 3 Proses 2.3.3 Wq

Aliran	Data
IN	Lq,a
OUT	Hsl_Wq
Proses	
Input Lq Hitung Wq $Wq = Hsl_Lq / a$	

Wq = Wq1 + Wq
 Output Hsl Wq

Tabel 2.14. PSPEC Level 3 Proses 2.3.4 idle

Aliran	Data
IN	N,Hsl_Lq, Hsl_Ls
OUT	Hsl_idle
Proses	
Input N Input Hsl_Lq Input Hsl_Ls Hitung idle = s (Ls – Lq) >= N Idle = idle1 + idle Output Dt_Idle	

Tabel 2.15. PSPEC Level 3 Proses 2.3.5 Stdr Deviasi

Aliran	Data
IN	N, Dt_idle
OUT	Dt_stdrDeviasi
Proses	
Input N Input Dt_Idle Hitung Stdr Deviasi Output dt_stdrDeviasi	

Tabel 2.16. PSPEC Level 3 Proses 2.3.6 Hsl Hitung

Aliran	Data
IN	Tmp_Hitung
OUT	Dt_UP
Proses	
Input Tmp_Hitung Add Tmp_Hitung Output Dt_UP	