

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai analisis dari permasalahan yang diambil beserta rancangan sistem dari Aplikasi Simulasi Pelayanan Restoran Cepat Saji Dengan Menggunakan Metode *Next-event Time Advance*.

3.1 Analisis Sistem

3.1.1 Identifikasi Permasalahan

Restoran cepat saji sangat mengutamakan kecepatan dalam pelayanannya. Pelayanan yang dimaksud adalah ketika pelanggan datang ke *counter* memesan makanan dan dilayani oleh pelayan restoran cepat saji tersebut. Namun seiring dengan berjalannya waktu dan bertambahnya konsumen, masalah demi masalah mulai timbul.

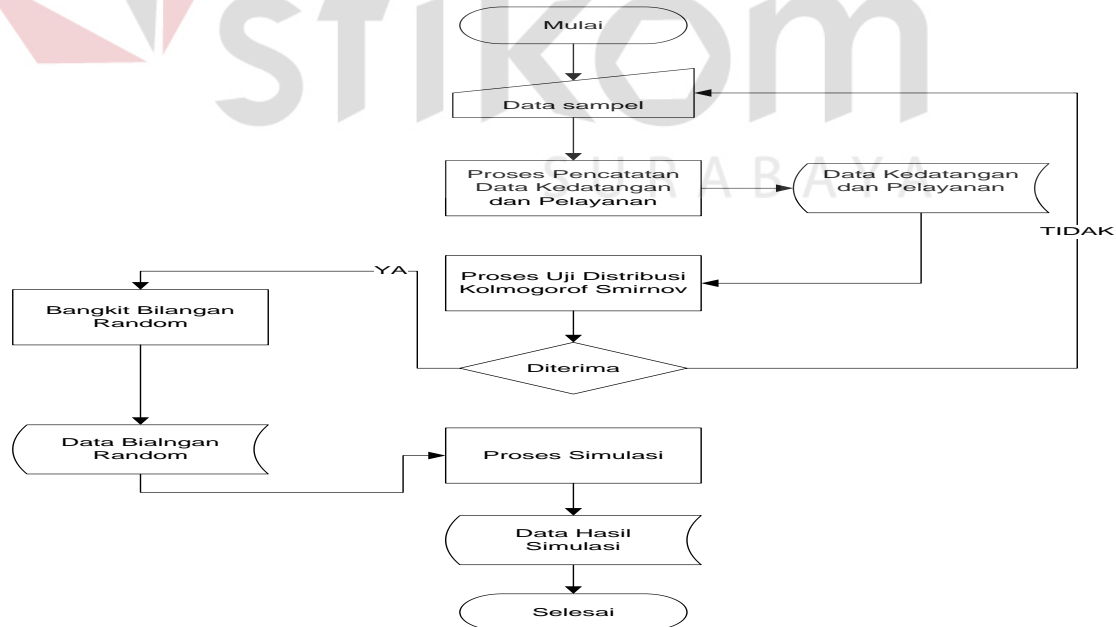
Mulai dari permasalahan lamanya konsumen dilayani yang mengakibatkan waktu tunggu konsumen lain untuk mendapatkan pelayanan menjadi semakin lama, permasalahan lain seperti kapan harus membuka *counter* dengan jumlah yang sedikit ketika kondisi restoran tidak begitu ramai, hingga sampai permasalahan kapan waktu yang tepat membuka *counter* baru untuk kondisi restoran yang ramai yang mengakibatkan panjangnya antrian. Kesalahan dalam memperkirakan jumlah *counter* yang akan dibuka di restoran berdampak sangat besar bagi keefektifan pelayanan dan juga keefisienan restoran dalam hal biaya yang akan dikeluarkan untuk membuka

setiap *counter*. Kesulitan ini nantinya akan mempengaruhi tingkat keefektifan dan keefisienan dari pelayanan restoran tersebut.

3.1.2 Analisis Permasalahan

Dari permasalahan yang ada tersebut, maka berikut adalah analisis dari sistem yang dibuat penulis untuk menyelesaikan. Prosedur pengembangan merupakan rangkaian alur penyelesaian dengan metode *next-event time advance* yang sudah baku untuk penyelesaian program simulasi yang penulis buat, dan bukan hanya merupakan dasar yang menjadi acuan untuk mengembangkan program tetapi juga untuk memecahkan masalah sesuai dengan kebutuhan yang diperlukan pada saat penyelesaian program.

Hal-hal tersebut akan penulis tuangkan dalam bentuk diagram penelitian seperti gambar 3.1 berikut.



Gambar 3.1 Alur Sistem Penyelesaian Metode *Next-event Time Advanced*

Dalam hal memperkirakan jumlah *counter* ini pihak restoran membutuhkan data kedatangan konsumen dan juga data lamanya pelayanan konsumen. Namun kondisi kedatangan konsumen yang tidak tetap atau fluktuatif perwaktunya mengakibatkan pihak restoran kesulitan dalam hal menentukan jumlah *counter* yang akan dibuka pada saat itu. Oleh Sebab itu dalam tugas akhir ini dibuatlah sebuah aplikasi simulasi antrian pelayanan. Aplikasi simulasi antrian pelayanan ini menggunakan metode *next-event time advance*.

Data yang didapatkan untuk membuat aplikasi simulasi antrian pelayanan adalah data yang bersifat kontinu, maka untuk pengujian distribusinya digunakan distribusi *Kolmogorov-Smirnov Normal*. Setelah data tersebut diuji, maka program simulasi antrian akan membangkitkan bilangan acak dengan distribusi normal. Kemudian simulasi antrian dapat dijalankan dan hasilnya ditujukan untuk mendapatkan gambaran tentang pengambilan keputusan manajer restoran tentang kapan harus menambah atau mengurangi *counter* yang beroperasi.

3.2 Proses Perhitungan Metode *Next-Event Time Advance*

Waktu antar kedatangan merupakan selang waktu antar kedatangan pelanggan ke t dan ke $t+1$. Dari data waktu antar kedatangan ini akan dilakukan pendugaan sementara distribusi probabilitas waktu antar kedatangan dengan cara menghitung koefisiensi variasi data dari hasil pengamatan. Dengan pendugaan sementara distribusi waktu antar kedatangan seperti pada table 3.1 berikut:

Tabel 3.1 Distribusi Frekuensi Waktu Antar Kedatangan

| No. | Batas Bawah | Batas Atas | Fi | Xi | fiXi | $\bar{X} = \frac{\sum f_i X_i}{n}$ | X_i^2 | $f_i \cdot X_i^2$ | Simpangan Baku |
|-----|-------------|------------|-----|-----|------|------------------------------------|---------|-------------------|----------------|
| 1 | 10 | 25 | 7 | 22 | 154 | | 484 | 3388 | |
| 2 | 26 | 41 | 11 | 38 | 418 | | 1444 | 15884 | |
| 3 | 42 | 57 | 10 | 54 | 540 | | 2916 | 29160 | |
| 4 | 58 | 73 | 28 | 70 | 1960 | | 4900 | 137200 | |
| 5 | 74 | 89 | 18 | 86 | 1548 | | 7396 | 133128 | |
| 6 | 90 | 105 | 21 | 102 | 2142 | | 10404 | 218484 | |
| 7 | 106 | 121 | 5 | 118 | 590 | | 13924 | 69620 | |
| | | | 100 | 490 | 7352 | 73.52 | 41468 | 606864 | 25.88728 |

Simpangan baku yang diperoleh dari table diatas adalah 25.88728 dan hasil dari distribusi frekuensi di atas di uji dengan distribusi normal dan menghasilkan suatu nilai Z. Kemudian nilai Z akan dicocokkan dengan table distribusi normal seperti tampak pada table 3.2

Tabel 3.2 Uji Distribusi Normal Waktu Antar Kedatangan

| No. | Batas Bawah | Batas Atas | $Z = \frac{(X_i - \bar{X})}{S}$ | F(x) Dari Tabel Dis Normal |
|-----|-------------|------------|---------------------------------|----------------------------|
| 1 | 10 | 25 | -1.990167 | 0.0233 |
| 2 | 26 | 41 | -1.372103 | 0.0853 |
| 3 | 42 | 57 | -0.754038 | 0.2266 |
| 4 | 58 | 73 | -0.135974 | 0.4483 |
| 5 | 74 | 89 | 0.4820901 | 0.6844 |
| 6 | 90 | 105 | 1.1001543 | 0.8643 |
| 7 | 106 | 121 | 1.7182186 | 0.9564 |

Jika nilai F(X) telah didapatkan melalui tabel distribusi normal maka selanjutnya akan melakukan uji dengan kolmogorof smirnov seperti pada tabel 3.3 di bawah ini, dimana nilai dari frekuensi kumulatif didapat dari kumulatif nilai Fi atau nilai

frekuensi. Nilai $S(X)$ didapatkan dari nilai frekuensi kumulatif dibagi dengan \bar{X} atau jumlah data yang telah ada.

Tabel 3.3 Uji Distribusi Normal Waktu Antar Kedatangan

| No. | Batas Bawah | Batas Atas | Frekuensi Kumulatif | $S(X)$ | $F(X)-S(X)$ | $ F(X)-S(X) $ |
|-----|-------------|------------|---------------------|--------|-------------|---------------|
| 1 | 10 | 25 | 7 | 0.07 | -0.0467 | 0.0467 |
| 2 | 26 | 41 | 18 | 0.18 | -0.0947 | 0.0947 |
| 3 | 42 | 57 | 28 | 0.28 | -0.0534 | 0.0534 |
| 4 | 58 | 73 | 56 | 0.56 | -0.1117 | 0.1117 |
| 5 | 74 | 89 | 74 | 0.74 | -0.0556 | 0.0556 |
| 6 | 90 | 105 | 95 | 0.95 | -0.0857 | 0.0857 |
| 7 | 106 | 121 | 100 | 1 | -0.0436 | 0.0436 |

Dari tabel uji distribusi normal didapatkan nilai $|F(X) - S(X)|$ yang tertinggi (terbesar) kemudian bandingkan dengan nilai dari table Kolmogorov Smirnov.

Dengan tingkat kepercayaan $\alpha = 0.05$ dengan $n = 100$, maka:

1. Jika $T_{hitung} < W_{1-\alpha}$ = maka H_0 Gagal Tolak
2. Jika $T_{hitung} \geq W_{1-\alpha}$ = maka H_0 DiTolak

Dari perhitungan diatas $T_{hitung} = 0,1117 < W_{1-\alpha} = 0,134$, jadi hasil dari uji distribusi Normal ini H_0 Gagal Tolak (diterima). Maka dapat disimpulkan bahwa data waktu kedatangan konsumen berdistribusi Normal lalu akan dibangkitkan bilangan random sesuai dengan hasil uji distribusi yang telah diterima.

Setelah diketahui data telah berdistribusi normal, selanjutnya adalah membangkitkan bilangan acak dengan menggunakan data berdistribusi normal tersebut.

Cara pembangkitan bilangan acak dengan menggunakan perhitungan sebagai berikut:

$$U_i = n_i/m$$

Keterangan:

U_i = bilangan acak *uniform*

N_i = data ke i

m = konstanta modulus

Contoh:

Data bilangan acak ke-1 diperoleh dari data waktu kedatangan dibagi konstanta modulus.

Data waktu kedatangan ke-1 = 60

Konstanta modulus 1 (untuk At_i) = 124

Konstanta modulus 2 (untuk St_i) = 128

Jadi: $U_1 At_i = 60 / 124 = 0,46875$

$U_1 St_i = 60 / 128 = 0,46875$

$U_2 At_i = 53 / 124 = 0,41406$

$U_2 St_i = 53 / 128 = 0,42742$

$U_3 At_i = 20 / 124 = 0,15625$

$U_3 St_i = 20 / 128 = 0,16129$

$U_4 At_i = 60 / 124 = 0,46875$

$U_4 St_i = 60 / 128 = 0,48387$

$U_5 At_i = 28 / 124 = 0,21875$

$U_5 St_i = 28 / 128 = 0,22581$

$U_6 At_i = 15 / 124 = 0,11719$

$U_6 St_i = 15 / 128 = 0,12097$

$U_7 At_i = 27 / 124 = 0,21094$

$U_7 St_i = 27 / 128 = 0,21774$

$U_8 At_i = 10 / 124 = 0,07813$

$U_8 St_i = 10 / 128 = 0,08065$

$U_9 At_i = 65 / 124 = 0,50781$

$U_9 St_i = 65 / 128 = 0,52419$

$U_{10} At_i = 42 / 124 = 0,32813$

$U_{10} St_i = 42 / 128 = 0,33871, \text{dst.}$

Tabel 3.4 Pembangkitan Bilangan Acak

| No. | Data Waktu Kedatangan | Acak Untuk Ati | Acak Untuk STI |
|-----|-----------------------|----------------|----------------|
| 1 | 60 | 0.46875 | 0.48387 |
| 2 | 53 | 0.41406 | 0.42742 |
| 3 | 20 | 0.15625 | 0.16129 |
| 4 | 60 | 0.46875 | 0.48387 |
| 5 | 28 | 0.21875 | 0.22581 |
| 6 | 15 | 0.11719 | 0.12097 |
| 7 | 27 | 0.21094 | 0.21774 |
| 8 | 10 | 0.07813 | 0.08065 |
| 9 | 65 | 0.50781 | 0.52419 |
| 10 | 42 | 0.32813 | 0.33871 |
| 11 | 63 | 0.49219 | 0.50806 |
| 12 | 76 | 0.59375 | 0.61290 |
| 13 | 35 | 0.27344 | 0.28226 |
| 14 | 80 | 0.62500 | 0.64516 |
| 15 | 62 | 0.48438 | 0.50000 |
| 16 | 10 | 0.07813 | 0.08065 |
| 17 | 79 | 0.61719 | 0.63710 |
| 18 | 13 | 0.10156 | 0.10484 |
| 19 | 60 | 0.46875 | 0.48387 |
| 20 | 45 | 0.35156 | 0.36290 |
| 21 | 15 | 0.11719 | 0.12097 |
| 22 | 42 | 0.32813 | 0.33871 |
| 23 | 60 | 0.46875 | 0.48387 |
| 24 | 64 | 0.50000 | 0.51613 |
| 25 | 50 | 0.39063 | 0.40323 |
| 26 | 59 | 0.46094 | 0.47581 |
| 27 | 80 | 0.62500 | 0.64516 |
| 28 | 58 | 0.45313 | 0.46774 |
| 29 | 63 | 0.49219 | 0.50806 |
| 30 | 95 | 0.74219 | 0.76613 |
| 31 | 58 | 0.45313 | 0.46774 |
| 32 | 30 | 0.23438 | 0.24194 |
| 33 | 40 | 0.31250 | 0.32258 |
| 34 | 55 | 0.42969 | 0.44355 |
| 35 | 59 | 0.46094 | 0.47581 |
| 36 | 12 | 0.09375 | 0.09677 |
| 37 | 60 | 0.46875 | 0.48387 |
| 38 | 58 | 0.45313 | 0.46774 |
| 39 | 95 | 0.74219 | 0.76613 |
| 40 | 60 | 0.46875 | 0.48387 |
| 41 | 57 | 0.44531 | 0.45968 |
| 42 | 67 | 0.52344 | 0.54032 |
| 43 | 60 | 0.46875 | 0.48387 |
| 44 | 62 | 0.48438 | 0.50000 |
| 45 | 33 | 0.25781 | 0.26613 |
| 46 | 75 | 0.58594 | 0.60484 |
| 47 | 65 | 0.50781 | 0.52419 |
| 48 | 110 | 0.85938 | 0.88710 |
| 49 | 65 | 0.50781 | 0.52419 |
| 50 | 76 | 0.59375 | 0.61290 |
| 51 | 50 | 0.39063 | 0.40323 |
| 52 | 90 | 0.70313 | 0.72581 |
| 53 | 75 | 0.58594 | 0.60484 |
| 54 | 92 | 0.71875 | 0.74194 |
| 55 | 37 | 0.28906 | 0.29839 |
| 56 | 75 | 0.58594 | 0.60484 |
| 57 | 90 | 0.70313 | 0.72581 |
| 58 | 54 | 0.42188 | 0.43548 |
| 59 | 120 | 0.93750 | 0.96774 |
| 60 | 100 | 0.78125 | 0.80645 |
| 61 | 77 | 0.60156 | 0.62097 |
| 62 | 60 | 0.46875 | 0.48387 |
| 63 | 77 | 0.60156 | 0.62097 |
| 64 | 32 | 0.25000 | 0.25806 |
| 65 | 67 | 0.52344 | 0.54032 |
| 66 | 79 | 0.61719 | 0.63710 |
| 67 | 105 | 0.82031 | 0.84677 |
| 68 | 68 | 0.53125 | 0.54839 |
| 69 | 90 | 0.70313 | 0.72581 |
| 70 | 80 | 0.62500 | 0.64516 |
| 71 | 35 | 0.27344 | 0.28226 |
| 72 | 72 | 0.56250 | 0.58065 |
| 73 | 110 | 0.85938 | 0.88710 |
| 74 | 82 | 0.64063 | 0.66129 |
| 75 | 90 | 0.70313 | 0.72581 |
| 76 | 90 | 0.70313 | 0.72581 |
| 77 | 30 | 0.23438 | 0.24194 |
| 78 | 90 | 0.70313 | 0.72581 |
| 79 | 100 | 0.78125 | 0.80645 |
| 80 | 115 | 0.89844 | 0.92742 |
| 81 | 90 | 0.70313 | 0.72581 |
| 82 | 85 | 0.66406 | 0.68548 |
| 83 | 80 | 0.62500 | 0.64516 |
| 84 | 90 | 0.70313 | 0.72581 |
| 85 | 70 | 0.54688 | 0.56452 |
| 86 | 100 | 0.78125 | 0.80645 |
| 87 | 82 | 0.64063 | 0.66129 |
| 88 | 32 | 0.25000 | 0.25806 |
| 89 | 90 | 0.70313 | 0.72581 |
| 90 | 70 | 0.54688 | 0.56452 |
| 91 | 100 | 0.78125 | 0.80645 |
| 92 | 92 | 0.71875 | 0.74194 |
| 93 | 82 | 0.64063 | 0.66129 |
| 94 | 55 | 0.42969 | 0.44355 |
| 95 | 105 | 0.82031 | 0.84677 |
| 96 | 90 | 0.70313 | 0.72581 |
| 97 | 69 | 0.53906 | 0.55645 |
| 98 | 120 | 0.93750 | 0.96774 |
| 99 | 79 | 0.61719 | 0.63710 |
| 100 | 95 | 0.74219 | 0.76613 |

Setelah data bilangan acak telah dibangkitkan dengan distribusi normal, selanjutnya proses dimulai simulasi antrian pelayanan restoran cepat saji dilakukan dengan menentukan terlebih dahulu waktu pelayanan, jumlah fasilitas, dan maksimum waktu tunggu.

Contoh kebijakan perusahaan dalam pelayanan:

Waktu pelayanan = shift pagi (shift ada tiga: pagi, siang, dan malam).

Jumlah *counter* yang dibuka = 3 fasilitas (fasilitas akan akan diijinkan ditambah jika waktu tunggu *counter* melebihi maksimum waktu tunggu).

Maksimum waktu tunggu = 60.

Ketiga data tersebut akan digunakan sebagai inputan dalam proses simulasi antrian pelayanan restoran cepat saji ini. simulasi dimulai dengan waktu kedatangan pelanggan ke-1 (A_{t1}) mengecek kondisi *counter* 1 yaitu kosong atau sedang melayani pelanggan. Ketika kosong maka pelanggan ke-1 akan langsung dilayani di *counter* 1. Namun jika *counter* 1 sedang melayani, maka harus dicek lagi lamanya pelayanan *counter* tersebut melebihi maksimum waktu tunggu atau tidak melebihi maksimum waktu tunggu. Jika tidak melebihi waktu tunggu maksimum maka pelanggan akan dilayani dengan syarat harus menunggu pelanggan sebelumnya selesai dilayani. Sedangkan jika melebihi maksimum waktu tunggu maka pelanggan akan dilayani di *counter* baru (*counter* 2). Alur perhitungan tersebut digunakan sampai proses simulasi antrian pelayanan restoran cepat saji selesai dijalankan.

Tabel 3.5 Simulasi Restoran Cepat Saji

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | |
|----|-----|---------|-----|---------|---------|-----|-------------|---------|----|-----|-------------|---------|----|------|-------------|---------|----|------|-------------|---------|----|--|
| 1 | | | | | | | | | | | | | | | | | | | | | | |
| 2 | pel | Acak1 | Ah | Al | Acak2 | Sti | fasilitas 1 | | | | fasilitas 2 | | | | fasilitas 3 | | | | fasilitas 4 | | | |
| 3 | 1 | 0.46875 | 0 | 8:00:00 | 0.42742 | 120 | 8:00:00 | 8:02:00 | 0 | 0 | B2 | D2 | W2 | #2 | B3 | D3 | W3 | #3 | B4 | D4 | W4 | |
| 4 | 2 | 0.41406 | 60 | 8:01:00 | 0.16129 | 60 | | | | | 8:01:00 | 8:02:00 | 0 | 60 | | | | | | | | |
| 5 | 3 | 0.15625 | 10 | 8:01:10 | 0.48387 | 120 | | | | | | | | | 8:01:10 | 8:03:10 | 0 | 70 | | | | |
| 6 | 4 | 0.46875 | 60 | 8:02:10 | 0.22581 | 60 | 8:02:10 | 8:03:10 | 0 | 10 | | | | | | | | | | | | |
| 7 | 5 | 0.21875 | 10 | 8:02:20 | 0.12097 | 60 | | | | | 8:02:20 | 8:03:20 | 0 | 20 | | | | | | | | |
| 8 | 6 | 0.11719 | 10 | 8:02:30 | 0.21774 | 60 | | | | | | | | | | | | | 8:02:30 | 8:03:30 | 0 | |
| 9 | 7 | 0.21094 | 10 | 8:02:40 | 0.08065 | 60 | | | | | | | | | | | | | | | | |
| 10 | 8 | 0.07813 | 10 | 8:02:50 | 0.52419 | 120 | 8:03:10 | 8:05:10 | 20 | 0 | | | | | | | | | | | | |
| 11 | 9 | 0.50781 | 300 | 8:07:50 | 0.33871 | 120 | 8:07:50 | 8:09:50 | 0 | 180 | | | | | | | | | | | | |
| 12 | 10 | 0.32813 | 60 | 8:08:50 | 0.50806 | 120 | | | | | 8:08:50 | 8:10:50 | 0 | 330 | | | | | | | | |
| 13 | 11 | 0.49219 | 60 | 8:09:50 | 0.61290 | 120 | 8:09:50 | 8:11:50 | 0 | 0 | | | | | | | | | | | | |
| 14 | 12 | 0.59375 | 300 | 8:14:50 | 0.28226 | 60 | 8:14:50 | 8:15:50 | 0 | 180 | | | | | | | | | | | | |
| 15 | 13 | 0.27344 | 60 | 8:15:50 | 0.64516 | 120 | 8:15:50 | 8:17:50 | 0 | 0 | | | | | | | | | | | | |
| 16 | 14 | 0.62500 | 300 | 8:20:50 | 0.50000 | 120 | 8:20:50 | 8:22:50 | 0 | 180 | | | | | | | | | | | | |
| 17 | 15 | 0.48438 | 60 | 8:21:50 | 0.08065 | 60 | | | | | 8:21:50 | 8:22:50 | 0 | 1280 | | | | | | | | |
| 18 | 16 | 0.07813 | 10 | 8:22:00 | 0.63710 | 120 | | | | | | | | | 8:22:00 | 8:24:00 | 0 | 1130 | | | | |
| 19 | 17 | 0.61719 | 300 | 8:27:00 | 0.10484 | 60 | 8:27:00 | 8:28:00 | 0 | 250 | | | | | | | | | | | | |
| 20 | 18 | 0.10156 | 10 | 8:27:10 | 0.48387 | 120 | | | | | 8:27:10 | 8:29:10 | 0 | 280 | | | | | | | | |
| 21 | 19 | 0.46875 | 60 | 8:28:10 | 0.36290 | 120 | 8:28:10 | 8:30:10 | 0 | 10 | | | | | | | | | | | | |
| 22 | 20 | 0.35156 | 60 | 8:29:10 | 0.12097 | 60 | | | | | 8:29:10 | 8:30:10 | 0 | 0 | | | | | | | | |
| 23 | 21 | 0.11719 | 10 | 8:29:20 | 0.33871 | 120 | | | | | | | | | 8:29:20 | 8:31:20 | 0 | 320 | | | | |
| 24 | 22 | 0.32813 | 60 | 8:30:20 | 0.48387 | 120 | 8:30:20 | 8:32:20 | 0 | 10 | | | | | | | | | | | | |
| 25 | 23 | 0.46875 | 60 | 8:31:20 | 0.51613 | 120 | | | | | 8:31:20 | 8:33:20 | 0 | 70 | | | | | | | | |

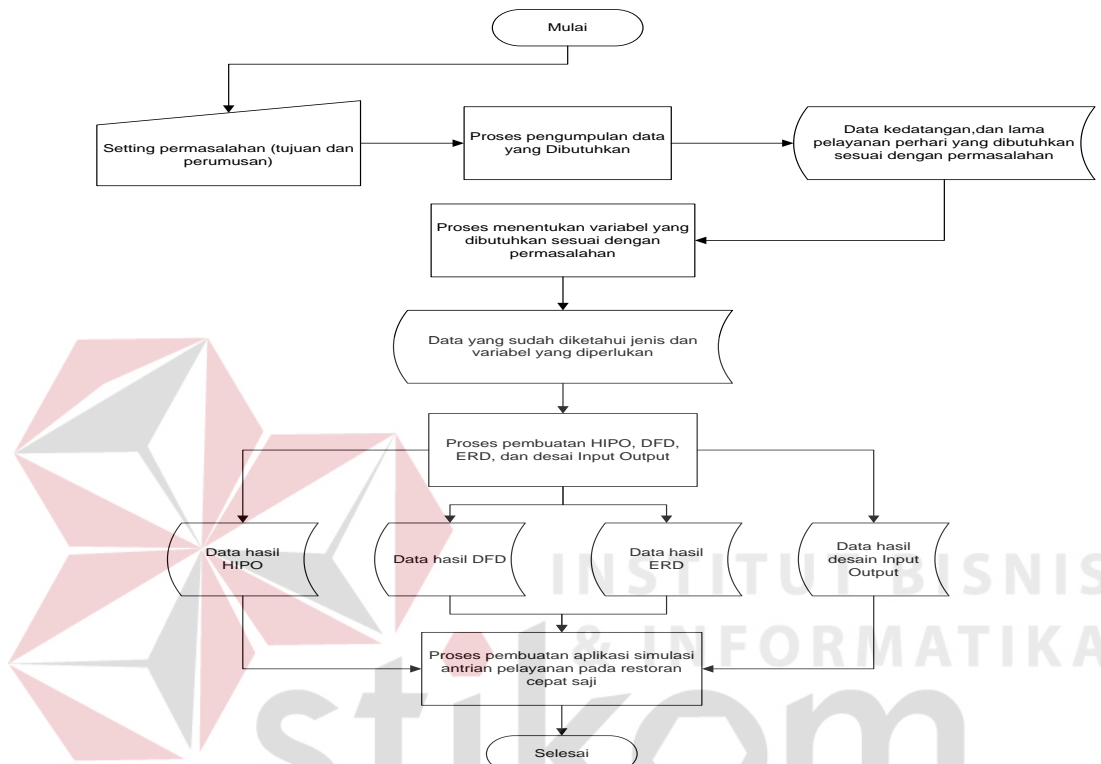
3.3 Perancangan Sistem

Penelitian yang dilakukan merupakan pembuatan program simulasi pengolahan data yang mewakili perilaku dari sistem yang benar-benar mengacu pada kejadian-kejadian yang nyata untuk pelayanan konsumen dan strategi penyediaan meja di restoran cepat saji.

Program simulasi ini bertujuan untuk memprediksikan jumlah *counter* pelayanan yang disiapkan pada hari dan segment waktu yang ditentukan.

Pada permasalahan tersebut di atas penulis mendapatkan beberapa event yang terjadi yaitu event kedatangan, dan event pelayanan. Setelah data setiap event sudah didapatkan, penulis akan menganalisa data tersebut dengan menerapkannya pada program simulasi tersebut sehingga penulis dapat melihat perilaku data pada setiap event yang ada.

Simulasi pelayanan restoran cepat saji ini menggunakan metode *next-event time advance*. Berikut ini adalah gambar blok Berikut gambar diagram penelitian program simulasi dengan menggunakan metode *next-event time advanced*.



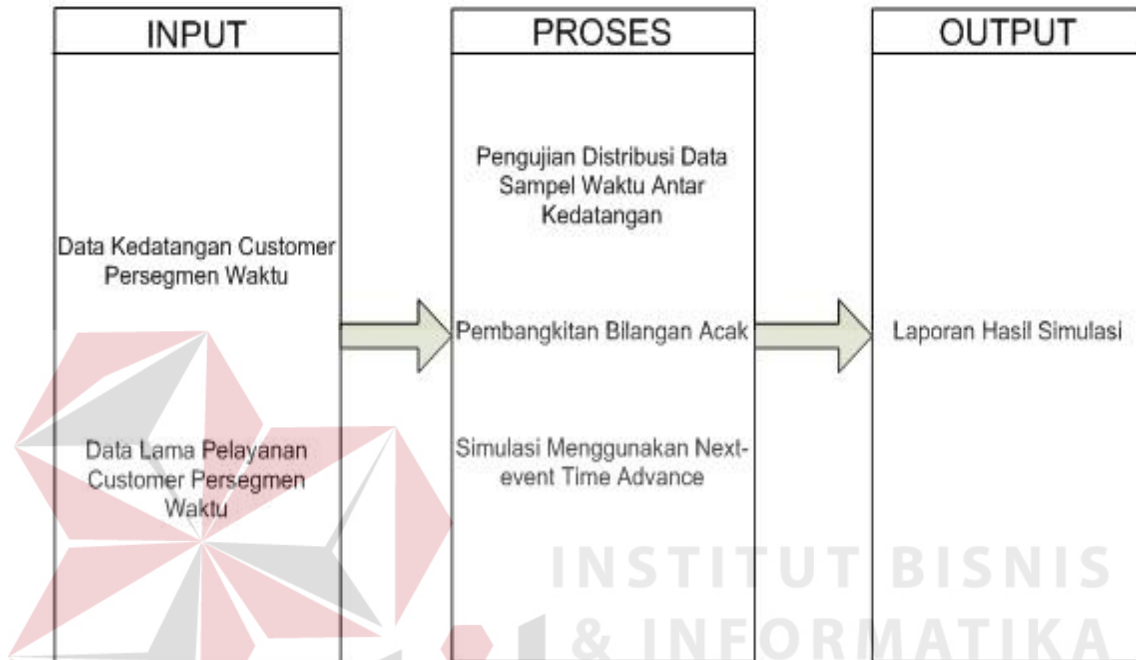
Gambar 3.2 Diagram Penelitian

Pada gambar 3.2 merupakan rangkaian dari alur yang di mulai dengan mengidentifikasi masalah, merumuskan masalah dan tujuan, mengumpulkan data, menentukan data yang dibutuhkan dan variabel untuk proses program simulasi, merancang sistem, dan lain-lain yang dibutuhkan hingga membuat laporan dari program simulasi tersebut.

Sedangkan program yang telah dijalankan memuat data-data yang berdistribusi countinu, maka memungkinkan bahwa penggunaan uji distribusi nantinya akan menggunakan distribusi kolmogorof-smirnov. Penulis dalam

mengembangkan program ini tidak menggunakan alat bantu apapun selain murni dari pengembangan rumus-rumus yang ada di buku teori statistik.

Model yang digunakan dalam diagram proses pelayanan yang akan dilakukan:



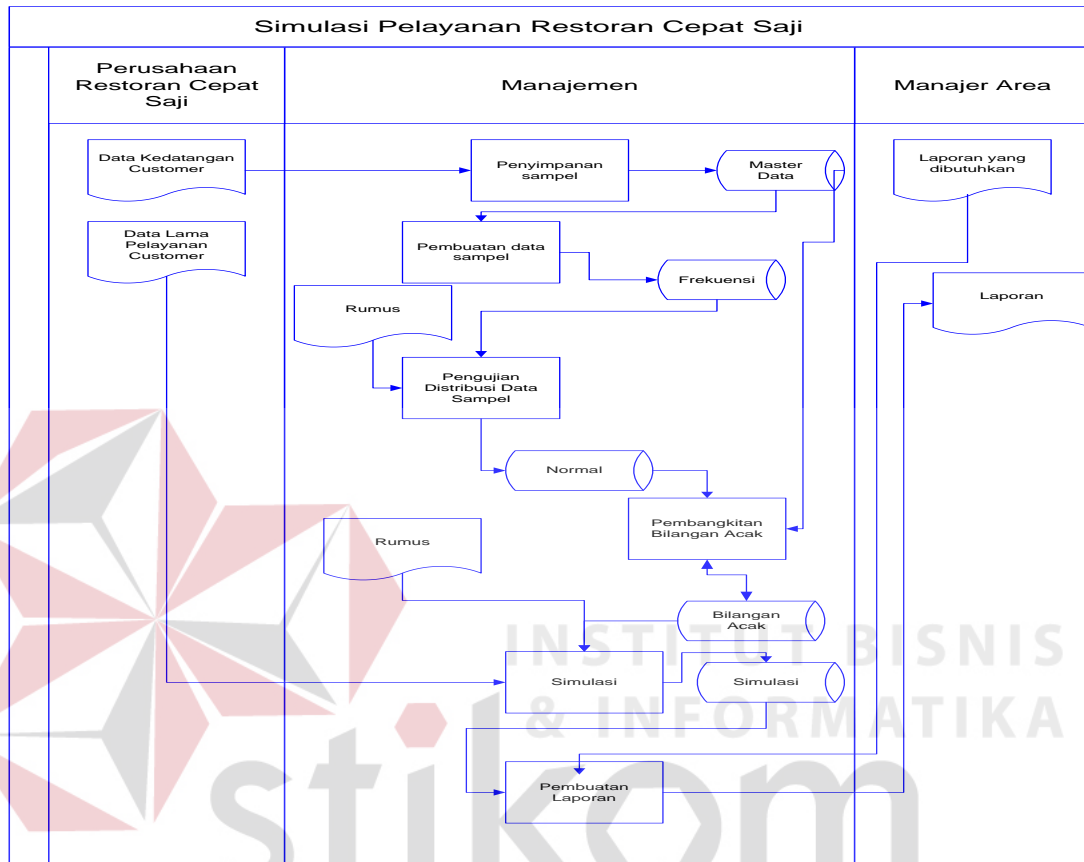
Gambar 3.3 Diagram Blok Simulasi Pelayanan Restoran Cepat Saji

3.3.1 System flow

Desain *System Data Flow Diagram* (DFD) menggunakan Power Desainer 6. *Proses Analyst*, untuk Desain Database digunakan Power Desainer 6. *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). Untuk *system flow* digunakan tool Microsoft Visio 2007.

Tugas akhir ini merupakan rancang bangun perangkat lunak, dibuat untuk membangun sebuah simulasi pelayanan yang dapat digunakan untuk menentukan jumlah *counter* pelayanan guna menunjang efisiensi dan efektifitas pelayanan

restoran tersebut dengan menggunakan metode *Next-Event Time Advance*. Alur proses sistem yang akan dibuat digambarkan sebagai berikut :

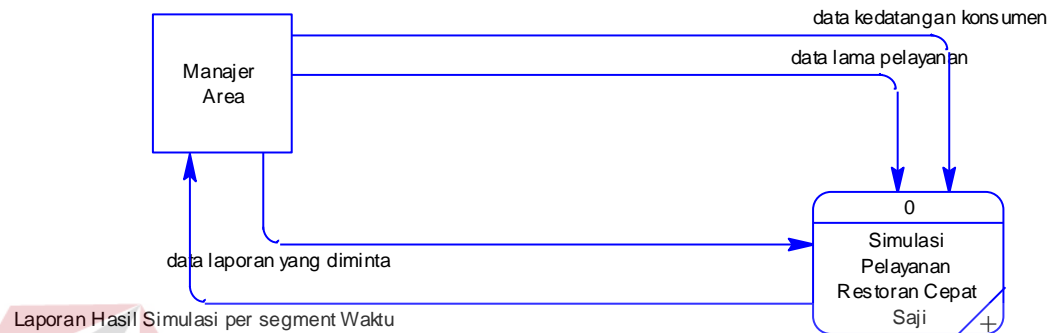


Gambar 3.4 *System Flow* Simulasi Pelayanan Restoran Cepat Saji

Berdasarkan gambar di atas dapat dilihat bahwa bagian manajemen menerima berbagai data yang diperlukan dalam melakukan simulasi. Seperti data kedatangan customer dan data lama pelayanan *customer*. Sehingga mendapatkan hasil akhir simulasi yang akan dibuat dalam bentuk laporan yang akan diberikan kepada pihak manajer area.

3.3.2 Context diagram

Context diagram pada untuk proses simulasi pelayanan restoran cepat saji dapat dilihat pada gambar di bawah ini.

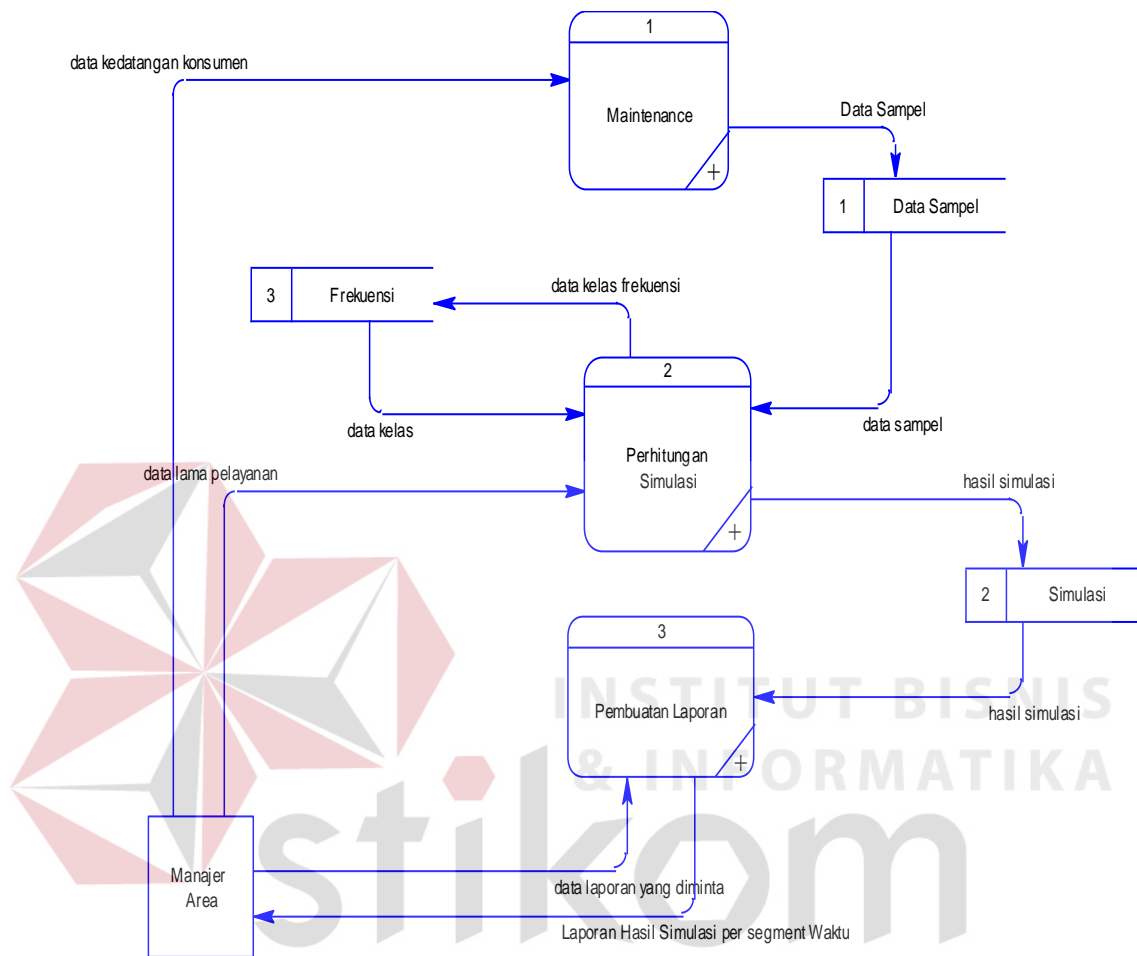


Gambar 3.5 *Context Diagram* Simulasi Pelayanan Restoran Cepat Saji

Dalam *context diagram* di atas, terdapat satu *entity* yaitu Manajer Area. *Context Diagram* ini dimulai dari Manajer Area yang menginputkan data berupa data kedatangan, dan lama pelayanan konsumen per segment waktu yang nantinya data-data ini diuji kenormalan distribusinya dengan uji distribusi normal. Setelah itu melakukan pembangkitan bilangan acak dan kemudian data tersebut disimulasikan. Dari data-data yang ada tersebut, digunakan oleh Manajer Area untuk melihat laporan berupa laporan hasil simulasinya yang nantinya akan digunakan untuk pengambilan keputusan oleh pihak Manajer Area.

Context Diagram ini akan dipecah lagi menjadi *data flow diagram* (DFD) level 0 ke dalam beberapa proses yaitu proses *maintenance*, penyimpanan data sampel, proses perhitungan frekuensi, dan pembuatan laporan.

3.3.3 Data Flow Diagram Level 0



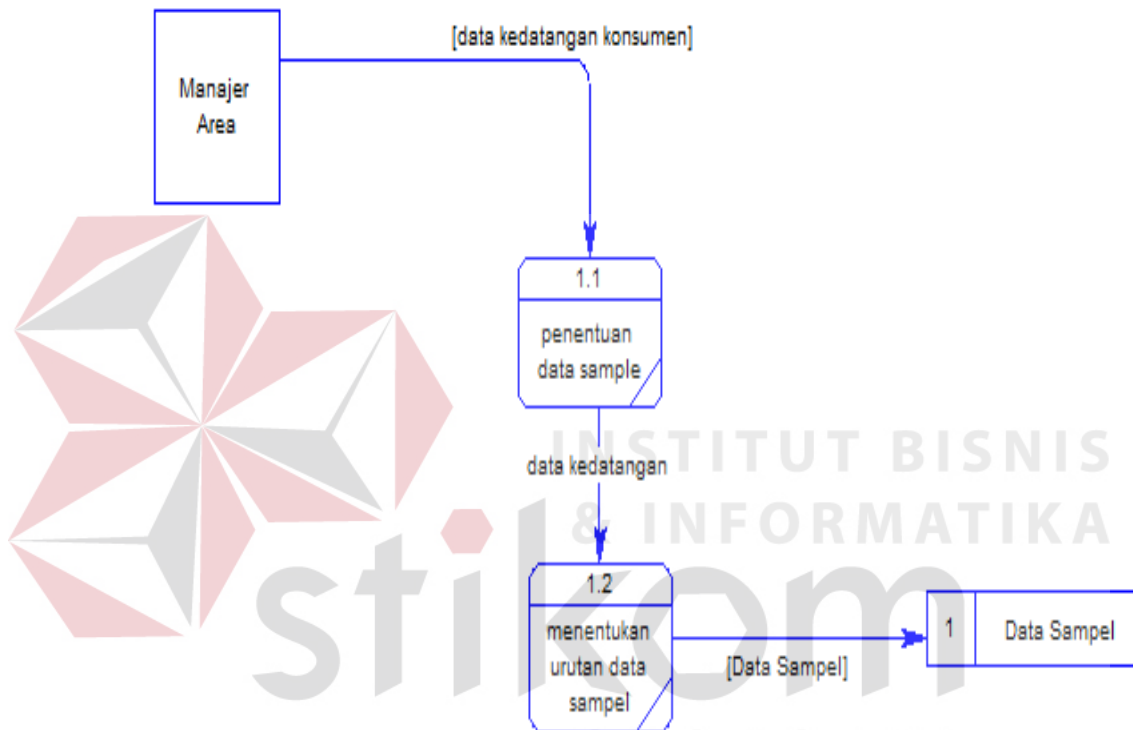
Gambar 3.6 Data Flow Diagram Level 0 Proses Simulasi Pelayanan Restoran Cepat Saji

Seperti yang telah dijelaskan diatas, DFD level 0 ini terdapat tiga proses yaitu proses *maintenance*, proses perhitungan simulasi dan pembuatan laporan. Pada DFD level 0 ini dimulai dari Manajer Area memberikan inputan data kedatangan konsumen yang akan akan diproses menjadi tabel data sampel dan tabel data rumus. kemudian dengan inputan data lama pelayanan, data sampel, data kelas, dan data rumus yang diproses dalam perhitungan simulasi akan menghasilkan data hasil simulasi. hasil simulasi yang disimpan ke dalam tabel simulasi digunakan untuk proses keempat

yaitu pembuatan laporan yang nantinya laporan tersebut yang berupa laporan hasil simulasi per segment waktu diterima oleh manager area.

3.3.4 Data Flow Diagram Level 1

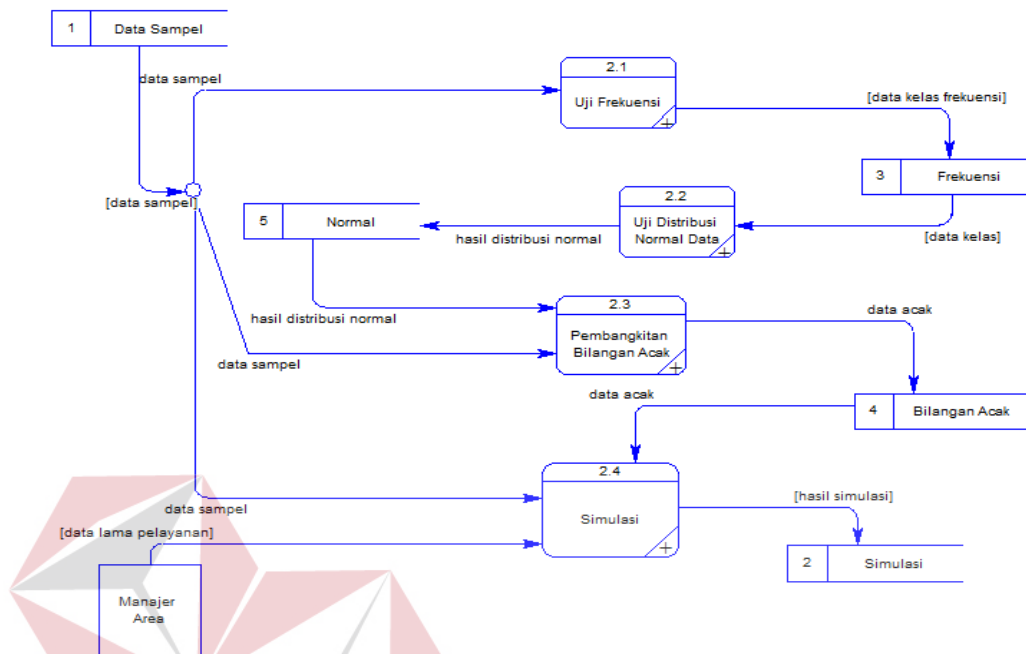
A. Data Flow Diagram Level 1 Proses Maintenance



Gambar 3.7 Data Flow Diagram Level 1 Proses Maintenance

Pada proses ini terdapat dua inputan dari Manager Area untuk dua proses yang ada. Inputan berupa data kedatangan konsumen digunakan dalam proses penentuan data sampel. Hasil dari proses proses menentukan urutan data sampel yang berupa data sampel akan disimpan ke dalam tabel data sampel.

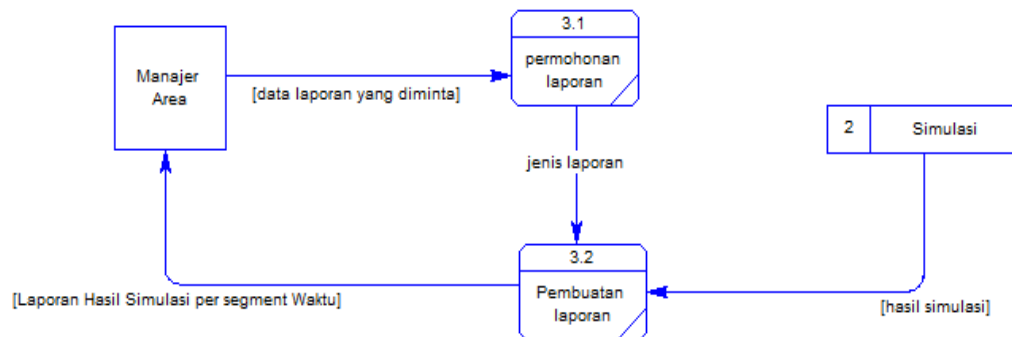
B. Data Flow Diagram Level 1 Proses Perhitungan Simulasi



Gambar 3.8 Data Flow Diagram Level 1 Proses Perhitungan Simulasi

Pada proses ini, terdapat empat proses yaitu proses uji frekuensi, uji distribusi normal, pembangkitan bilangan acak, dan simulasi. Proses dimulai data sampel yang digunakan sebagai inputan perhitungan pada proses uji frekuensi dan menghasilkan data kelas frekuensi yang kemudian disimpan kedalam tabel frekuensi. Data yang ada pada tabel frekuensi digunakan pada proses uji distribusi normal. Data tersebut digunakan untuk menguji sifat distribusinya. Dan hasil dari pengujian tersebut berupa data hasil distribusi normal dan kemudian disimpan kedalam tabel normal. Setelah itu data dari tabel normal digunakan pada proses pembangkitan bilangan acak dengan tambahan inputan data sampel dari tabel data sampel yang akan disimpan ke dalam tabel bilangan acak dan data tabel acak tersebut akan diolah pada proses simulasi untuk menentukan data hasil simulasi.

C. Data Flow Diagram Level 1 Proses Pembuatan Laporan

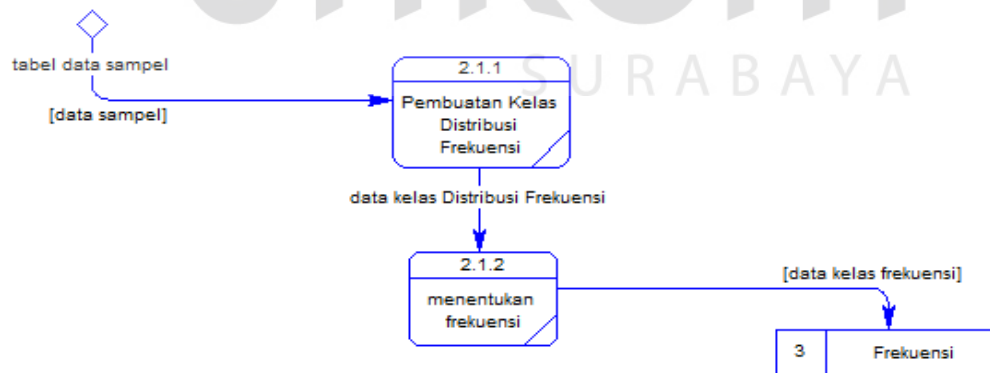


Gambar 3.9 Data Flow Diagram Level 1 Proses Pembuatan Laporan

Pada proses ini Manajer Area melakukan inputan data laporan yang diminta dan akan diproses pada proses permohonan laporan untuk menentukan jenis laporan yang dibutuhkan tersebut. Kemudian melakukan proses pembuatan laporan tersebut dan memberikannya kembali kepada Manajer Area.

3.3.5 Data Flow Diagram Level 2

A. Data Flow Diagram Level 2 Proses Uji Frekuensi

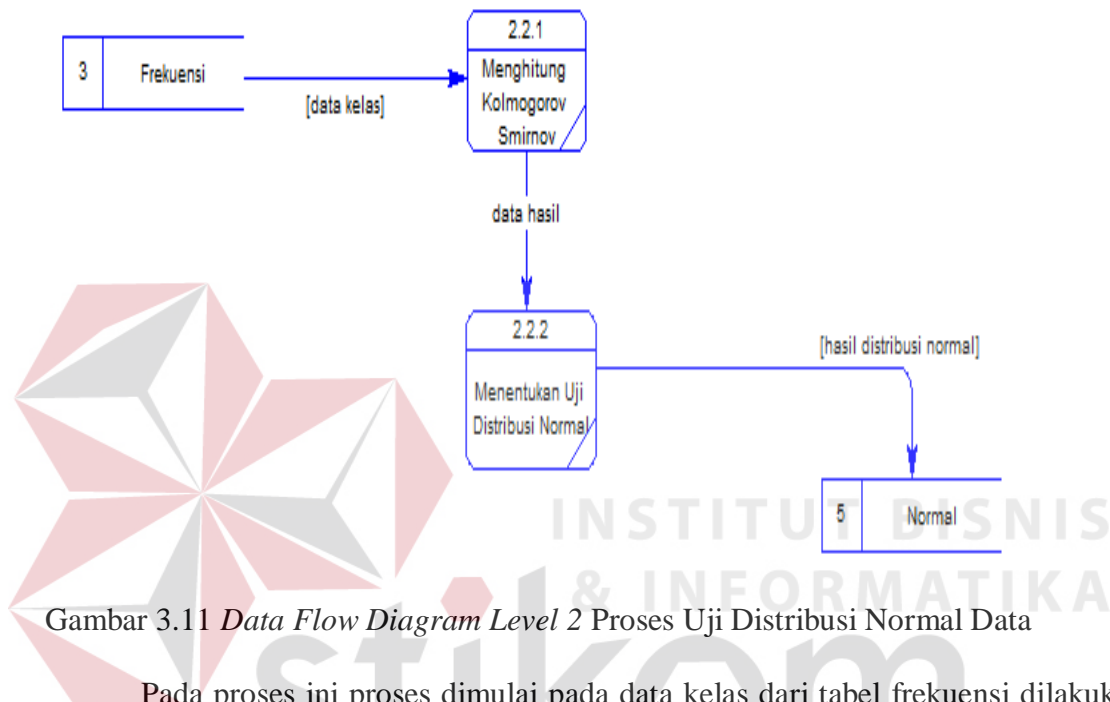


Gambar 3.10 Data Flow Diagram Level 2 Proses Uji Frekuensi

Pada gambar diatas, data sampel dari tabel data sampel diolah dalam proses pembuatan kelas distribusi frekuensi dengan perhitungan-perhitungan tertentu. Hasil

dari proses ini diproses dalam penentuan frekuensi untuk mencari nilai dari kelas distribusi frekuensinya dan kemudian disimpan ke dalam tabel frekuensi.

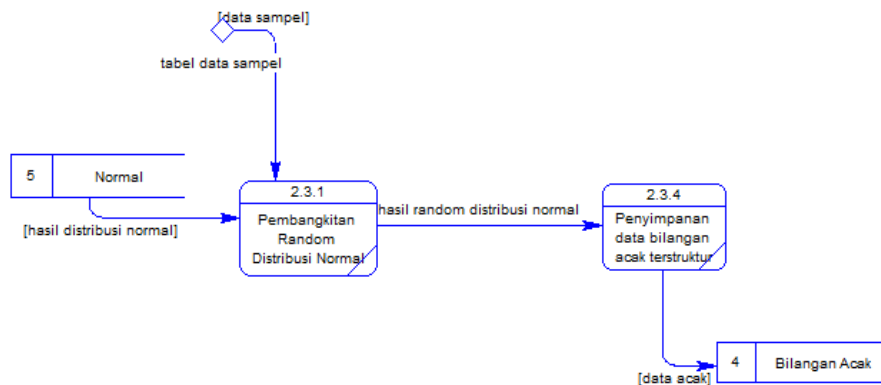
B. Data Flow Diagram Level 2 Proses Uji Distribusi Normal Data



Gambar 3.11 Data Flow Diagram Level 2 Proses Uji Distribusi Normal Data

Pada proses ini proses dimulai pada data kelas dari tabel frekuensi dilakukan perhitungan mencari nilai kolmogorov smirnov yang akan digunakan pada proses menentukan uji distribusi normal. Hasil dari proses menentukan uji distribusi normal ini berupa data hasil distribusi normal yang kemudian akan disimpan ke dalam tabel normal.

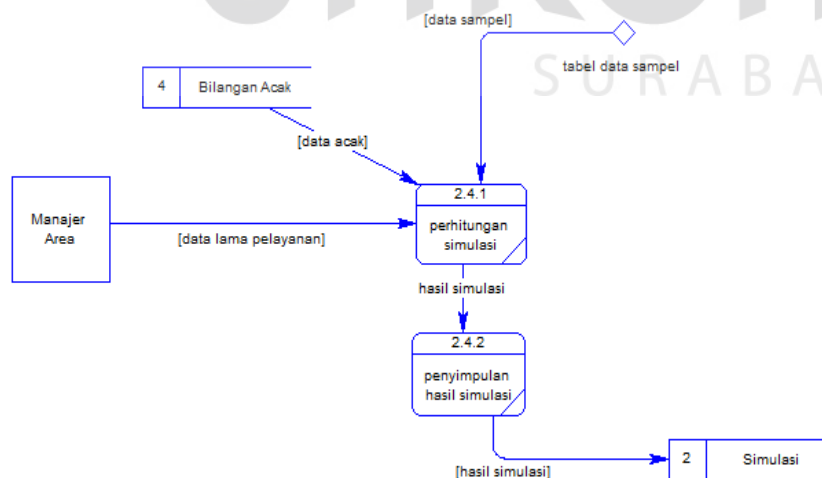
C. Data Flow Diagram Level 2 Proses Pembangkitan Bilangan Acak



Gambar 3.12 Data Flow Diagram Level 2 Proses Pembangkitan Bilangan Acak

Proses awal dimulai dari data dari data sampel diproses dengan perhitungan-perhitungan untuk pembangkitan bilangan acak sesuai dengan kebutuhan dalam proses simulasi. Hasil dari masing-masing proses pembangkitan bilangan acak diolah melalui proses penyimpanan dan hasilnya disimpan ke dalam tabel acak.

D. Data Flow Diagram Level 2 Proses Simulasi



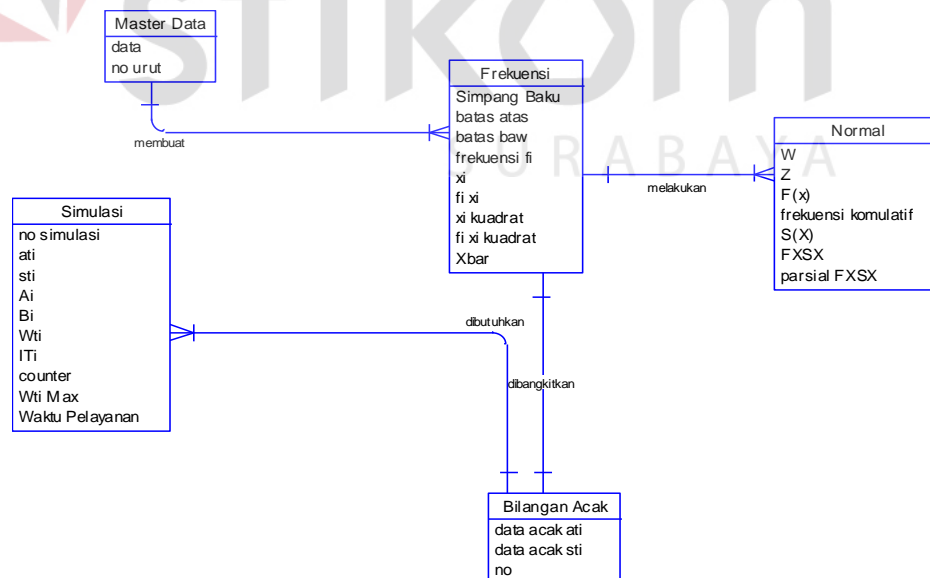
Gambar 3.13 Data Flow Diagram Level 2 Proses Simulasi

Pada gambar diatas, data dari tabel bilangan acak, data lama pelayanandan data sampel diolah dalam proses perhitungan simulasi dengan perhitungan-perhitungan tertentu. Dan hasil dari proses ini lakukan penyimpulan kedalam proses dalam proses penyimpulan hasil simulasi untuk disimpan ke dalam tabel simulasi.

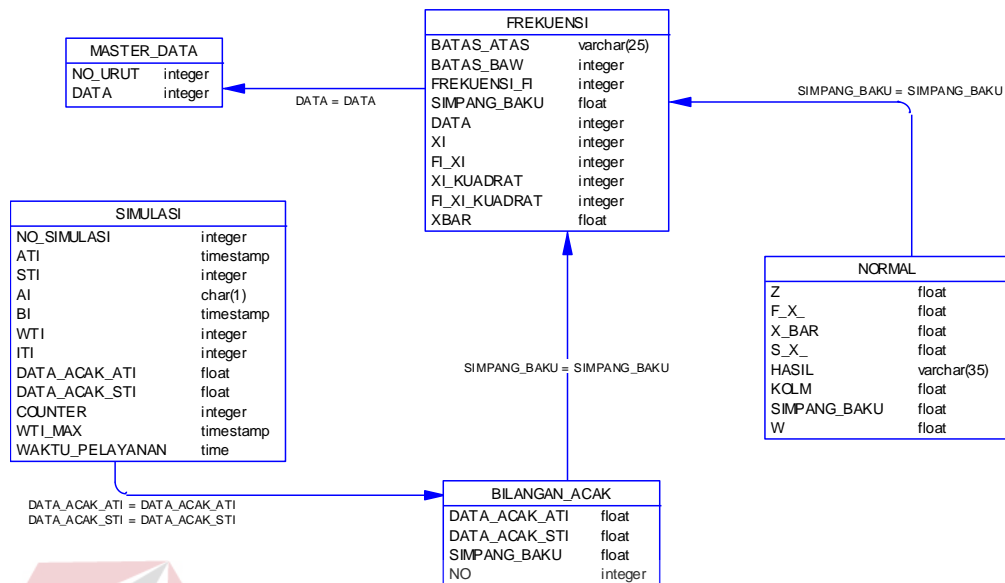
3.4 Entity Relationship Diagram

Entity Relationship Diagram digunakan untuk menginterpretasikan, menentukan, dan mendokumentasikan kebutuhan-kebutuhan untuk pemrosesan sistem *database* serta menunjukkan *relationship* dari beberapa data dalam *entity* yang saling berhubungan dalam sistem.

Berikut adalah *entity relationship diagram* yang dibuat dalam merancang *system database* yang terdapat dalam gambar *conceptual data modelling* (CDM) dan *physical data modelling* (PDM).



Gambar 3.14 *Conceptual Data Modelling*



Gambar 3.15 Physical Data Modelling

3.5 Struktur Tabel

Dalam hal merancang struktur tabel yang diperlukan, meliputi nama tabel, nama *atribut*, tipe data pelengkap seperti *primary key*. Rancangan basis data aplikasi ini menggunakan data base *Microsoft SQL Server 2007* dan terdiri dari tabel-tabel sebagai berikut:

1) Tabel Data Sampel

Nama Tabel : Data Sampel

Fungsi : Untuk menyimpan data sampel atau data dasar untuk kebutuhan simulasi

Tabel 3.6 Data Sampel

| Kolom | Panjang | Tipe | Keterangan |
|--------|---------|---------|------------|
| Nourut | 3 | Integer | |
| Data | 3 | Integer | PK |

2) Tabel Frekuensi

Nama Tabel : Frekuensi

Fungsi : Untuk menyimpan data distribusi frekuensi

Tabel 3.7 Frekuensi

| Kolom | Panjang | Tipe | Keterangan |
|---------------|---------|---------|------------|
| Simpang Baku | 4 | Float | PK |
| Data | 3 | Integer | FK |
| Batas Atas | | Integer | |
| Batas Bawah | | Integer | |
| Xi | | Integer | |
| Fi | | Integer | |
| Fi.Xi | | Integer | |
| Xi Kuadrat | | Integer | |
| Fi.Xi Kuadrat | | Integer | |
| Xbar | 4 | Float | |

3) Tabel Acak

Nama Tabel : Acak

Fungsi : Untuk menyimpan bilangan acak

Tabel 3.8 Acak

| Kolom | Panjang | Tipe | Keterangan |
|---------------|---------|---------|------------|
| Data Acak Ati | 4 | Float | PK |
| Data Acak Sti | 4 | Float | |
| No | | Integer | |
| Data | 3 | Integer | FK |

4) Tabel normal

Nama Tabel : Normal

Fungsi : Untuk menyimpan data hasil perhitungan distribusi normal

Tabel 3.9 Normal

| Kolom | Panjang | Tipe | Keterangan |
|---------------------|---------|---------|------------|
| W | 4 | Float | |
| Simpang Baku | 4 | Float | FK |
| Z | 4 | Float | |
| F(x) | 4 | Float | |
| Frekuensi Kumulatif | | Integer | |
| S(x) | 4 | Float | |
| F(x)-S(x) | 4 | Float | |
| F(x)-S(x) | 4 | Float | |

5) Tabel Simulasi

Nama Tabel : Simulasi

Fungsi : Untuk menyimpan data hasil simulasi

Tabel 3.10 Simulasi

| Kolom | Panjang | Tipe | Keterangan |
|-----------------|---------|-----------|------------|
| No Simulasi | | Integer | PK |
| Data Acak Ati | 4 | Float | FK |
| Ati | 4 | Timestamp | |
| Sti | 4 | Timestamp | |
| Ai | | Time | |
| Bi | | Time | |
| Wti | 4 | Timestamp | |
| ITi | 4 | Timestamp | |
| Counter | | Integer | |
| Wti Max | 3 | Timestamp | |
| Waktu Pelayanan | | Time | |

3.6 Desain Input Output

Desain *Input Output* (I/O) merupakan rancangan awal dari pembuatan suatu aplikasi. Desain I/O sangat membantu dalam mempercepat proses pengembangan serta mengetahui dengan lebih jelas alur program yang akan di rancang. Desain I/O

menggambarkan desain sistem dalam bentuk form-form yang mengimplementasikan inputan dan rancangan *output* yang berupa laporan-laporan dimana laporan-laporan tersebut digunakan sebagai dokumentasi. Desain I/O untuk aplikasi simulasi pelayanan restoran cepat saji dengan menggunakan metode *next-event time advanced* dapat digambarkan sebagai berikut:

3.6.1 Desain form utama

The image shows a wireframe of a software application window titled "Applikasi Simulasi Antrian". On the left side, there is a vertical menu bar containing the following elements from top to bottom: a "MENU" header, three buttons labeled "Waktu Pelayanan", "Data Waktu Kedatangan", and "Simulasi", and two more buttons labeled "Utility" and "Menu". The main area of the window is mostly blank, overlaid with a large, semi-transparent watermark logo for "stikom SURABAYA" (Institut Bisnis & Informatika Surabaya).

Gambar 3.16 Desain Form Utama

Pada gambar 3.16 di atas merupakan desain form utama. Form utama adalah form yang pertama kali muncul ketika aplikasi dijalankan. Pada form ini terdapat menu yang merupakan navigasi untuk mengakses form-form berikutnya. Navigasi

menu terdapat di bagian sebelah kiri. Navigasi menu terbagi menjadi 4 bagian yaitu Utility, Waktu Pelayanan, Data Waktu Kedatangan, dan Simulasi.

3.6.2 Desain form master waktu pelayanan

Applikasi Simulasi Antrian

MENU

Waktu Pelayanan

Data Waktu Kedatangan

Simulasi

Utility

Menu

| | Nama | Jam Awal | Jam Akhir |
|--|------|----------|-----------|
| | | | |
| | | | |
| | | | |
| | | | |

Data Baru Simpan Data Hapus Data Keluar

Nama

Jam Awal Jam Akhir

Rentang Acak Ati

Rentang Acak Sti

Gambar 3.17 Desain Form Master Waktu Pelayanan

Pada gambar 3.17 di atas merupakan form master waktu pelayanan. Form master waktu pelayanan ini berfungsi untuk melakukan manajemen terhadap data waktu pelayanan yang dimiliki oleh perusahaan. Pada form ini pengguna dapat memasukkan data waktu pelayanan baru, melakukan perubahan terhadap data waktu

pelayanan yang ada, ataupun menghapus data waktu pelayanan. Terdapat beberapa isian yang diperlukan yaitu nama waktu pelayanan (berdasarkan pembagian jam kerja), jam awal, jam akhir, rentang acak, Ati (indeks waktu kedatangan pelanggan), dan Sti (indeks waktu lama pelayanan). Data yang telah tersimpan akan terlihat pada tabel di sebelah kiri, dan apabila akan dilakukan perubahan data atau menghapus data dapat dilakukan dengan memilih data pada tabel tersebut.

3.6.3 Desain form Master Waktu Kedatangan

The screenshot shows a web-based form titled 'Applikasi Simulasi Antrian'. On the left is a vertical menu with buttons for 'Waktu Pelayanan', 'Data Waktu Kedatangan', 'Simulasi', 'Utility', and 'Menu'. The central part of the form contains a table with two columns: 'Keterangan' and 'Jumlah Data'. To the right of the table are four buttons: 'Data Baru', 'Simpan Data', 'Hapus Data', and 'Keluar'. Below these buttons are input fields for 'Nama' and 'Data', with an 'Import Excel' button next to the 'Nama' field. A large watermark for 'stikom SURABAYA' is visible across the center of the interface.

Gambar 3.18 Desain Form Master Waktu Kedatangan

Pada gambar 3.18 di atas merupakan form master waktu kedatangan. Form master waktu kedatangan berfungsi untuk melakukan manajemen terhadap data waktu kedatangan yang dimiliki oleh perusahaan. Pada form ini pengguna dapat memasukkan data waktu kedatangan baru, melakukan perubahan terhadap data waktu kedatangan yang ada, ataupun menghapus data waktu kedatangan. Terdapat dua cara

