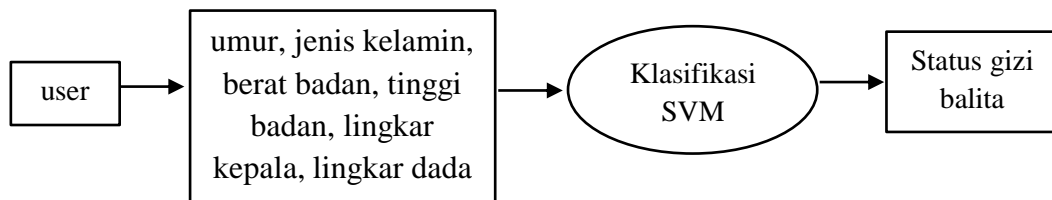


## BAB IV

### ANALISIS DAN DESAIN SISTEM

#### 4.1. Analisis Sistem

Tahapan analisis sistem ini, mempunyai tugas mendefinisikan masalah sistem, melakukan studi kelayakan, menganalisis kebutuhan sistem dan elemen-elemen yang terkait seperti *user* dan semua yang diperlukan dalam proses klasifikasi status gizi balita di Desa Banjaragung Kecamatan Bareng Jombang.



Pada kali ini dimulai dari *user* (pengguna) untuk *login* terlebih dahulu, kemudian setelah masuk di halaman perhitungan klasifikasi *user* harus mengisi sesuai list data yang dimiliki, setelah itu pilih tombol submit maka program akan otomatis menghitung data yang ada dengan klasifikasi metode SVM dan akan muncul hasil status gizi balita dari perhitungan tersebut.

##### 4.1.1. Analisis Masalah

Setiap kader desa tentunya ingin meningkatkan kualitasnya. salah satu caranya adalah dengan menjadikan balita yang sehat. Di Desa Banjaragung, Kecamatan Bareng Jombang, tingkat kesehatan setiap balita mempunyai perbedaan. Hal ini dapat dipengaruhi oleh beberapa faktor diantaranya : perekonomian keluarga, pola makanan yang dikonsumsi, pola asuh orang tua, pengaruh lingkungan, dan Gen (keturunan). Perlu adanya sebuah terobosan baru untuk merancang sebuah sistem klasifikasi gizi balita yang dijadikan pihak kader desa sebagai dasar klasifikasi dengan memanfaatkan data balita.

Dengan memanfaatkan sistem *data mining* khususnya metode SVM dibuat sebuah perancangan aplikasi untuk mengklasifikasikan gizi balita berdasarkan data dari kader desa. Ada dua kategori status gizi pada balita

yaitu gizi baik (normal) dan gizi kurang baik (gizi buruk). Dengan adanya aplikasi ini diharapkan dapat dimanfaatkan sebagai *user* yaitu kader desa untuk klasifikasi gizi pada balita.

#### 4.1.2. Analisis Data

Data yang digunakan dalam penulisan tugas akhir ini adalah data kuantitatif dari posyandu desa, dalam mempertimbangkan status gizi balitanya petugas kesehatan yang ada di posyandu Desa Banjaragung menggunakan empat parameter tetap yaitu umur, jenis kelamin, berat badan, tinggi badan. Namun yang akan dibuat oleh sistem menggunakan enam parameter yaitu umur, jenis kelamin, berat badan dan tinggi badan, lingkar kepala juga lingkar dada

##### 4.1.2.1. Sumber Data

Pada posyandu Desa Banjaragung terdapat 314 Data balita yang akan digunakan untuk uji data *training* dan uji data *testing*. Pada pengujian kali ini dibagi menjadi dua bagian jumlah data yaitu 70% data *training* – 30% data *testing* atau bisa juga 50% data *training* – 50% data *testing*. Hal ini didasarkan pada kebutuhan data yang akan dihubungkan pada sistem klasifikasi status gizi balita. Data yang diambil adalah laporan balita yang ada di Desa Banjaragung dengan menggunakan kriteria umur, jenis kelamin, berat badan, tinggi badan, lingkar kepala, lingkar dada merupakan atribut biner yaitu nominal yang hanya berisi nilai saja 0 atau 1, dimana 0 berarti atribut tidak ada, dan 1 berarti atribut ada.

Table 4.1 Tabel Status Gizi Balita

Status Gizi Balita	Nilai
Normal	1
Kurang Gizi	0

Tabel 4.1 merupakan tabel status gizi balita yang ada di Desa Banjaragung Kecamatan Bareng Jombang. Ada dua jenis status gizi balita yaitu normal dan kurang gizi. Normal memiliki nilai 1 dan kurang gizi memiliki nilai 0.

Table 4.2 Tabel Jenis Kelamin

Jenis Kelamin	jk1	jk2
Laki-Laki	1	0
Perempuan	0	1

Tabel 4.2 merupakan tabel jenis kelamin yang ada di Desa Banjaragung Kecamatan Bareng Jombang. Jenis kelamin tersebut meliputi laki-laki (jk1) dan perempuan (jk2).

Table 4.3 Tabel Berat Badan

Berat Badan	bb1	bb2
Naik	1	0
Turun	0	1

Tabel 4.3 merupakan tabel Berat Badan balita yang ada di Desa Banjaragung Kecamatan Bareng Jombang. Berat Badan tersebut meliputi naik (bb1) dan turun (bb2).

Table 4.4 Tabel Tinggi Badan

Tinggi Badan	tb1	tb2
Tinggi	1	0
Rendah	0	1

Tabel 4.4 merupakan tabel Tinggi Badan balitayang ada di Desa Banjaragung Kecamatan Bareng Jombang. Tinggi Badan tersebut meliputi Tinggi (tb1) dan Rendah (tb2).

Table 4.5 Tabel Lingkar Kepala

Lingkar Kepala	lk1	lk2
Besar	1	0
Kecil	0	1

Tabel 4.5 merupakan tabel Lingkar Kepala balita yang ada di Desa Banjaragung Kecamatan Bareng Jombang. Lingkar Kepala tersebut meliputi Besar (lk1) dan Kecil (lk2).

Table 4.6 Tabel Lingkar Dada

Lingkar Dada	ld1	ld2
Gemuk	1	0
Kurus	0	1

Tabel 4.6 merupakan tabel Lingkar Dada balita yang ada di Desa Banjaragung Kecamatan Bareng Jombang. Lingkar Dada tersebut meliputi Gemuk (ld1) dan Kurus (ld2).

## 4.2. Perancangan Sistem

Sistem yang akan dirancang dibagi menjadi lima proses yaitu proses normalisasi, proses *input data*, *preprocessing*, proses *training*, dan proses *testing*. Setiap masing-masing proses akan dijelaskan pada poin 4.2.1 sampai dengan 4.2.5 sebagai berikut :

### 4.2.1. Proses Normalisasi

Normalisasi adalah proses pengelompokan atribut data untuk meminimalkan redundansi data pada suatu database agar dapat bekerja dengan *optimal*. Rumus yang digunakan pada normalisasi ini adalah :

$$\text{New data} = (\text{nilai X} - \text{X min}) / (\text{X max} - \text{X min})$$

X Min = angka terkecil dari seluruh variabel data.

X Max = angka terbesar dari seluruh variabel data.

Tabel 4.7 contoh normalisasi data

No	U	Norm U	BB	Norm BB	TB	Norm TB	LK	Norm Lk	LD	Norm LD
1	4	0.75	12	0.6	89	0.625	43	0.643	42	0.643
2	2	0.25	12	0.6	82	0.450	39	0.357	39	0.429
3	1	0	6	0	64	0.000	34	0.000	33	0.000
4	5	1	16	1	104	1.000	48	1.000	47	1.000
5	1	0	7	0.1	75	0.275	34	0.000	33	0.000
6	3	0.5	10	0.4	90	0.650	42	0.571	41	0.571
7	5	1	16	1	102	0.950	48	1.000	47	1.000
8	3	0.5	9	0.3	78	0.350	43	0.643	42	0.643
9	2	0.25	10	0.4	79	0.375	43	0.643	42	0.643
10	4	0.75	13	0.7	100	0.900	44	0.714	43	0.714

Tabel 4.7 diatas terdapat 10 nilai data asli umur, berat badan, tinggi badan, lingkar kepala, dan ingkar dada, juga hasil dari normalisasi sesuai nilai variabel data yang dihitung menggunakan rumus normalisasi diatas.

#### 4.2.2. Proses Input Data

Didalam proses *input* data yaitu proses untuk memasukkan, mengubah atau menghapus data-data oleh sistem diantaranya :

Data *input* yang dimasukkan adalah data yang dimasukkan oleh *user*, yang kemudian diolah dengan sistem untuk menghasilkan data baru, sedangkan data *output* adalah data yang diinginkan oleh *user* untuk ditampilkan atau diselesaikan oleh sistem sesuai dengan data masukkan sebelumnya.

1. Data *input* yang diperlukan program adalah :
  - a) Data balita untuk data pelatihan.
  - b) Data balita untuk data pengujian.
2. Data keluar yang dihasilkan oleh program aplikasi adalah :  
 Hasil pengklasifikasian dari data *testing* yang diselesaikan oleh sistem dimana menghasilkan prediksi keberadaan kelas manakah yang lebih tepat untuk data tersebut.

#### 4.2.3. Proses Preprocessing

Pada tahap ini data yang sudah dimasukkan akan dinormalisasikan sesuai nilai variabelnya berdasarkan atribut dari data tersebut. Contoh data yang sebelum dinormalisasi dan sudah ternormalisasi dapat dilihat pada tabel 4.7 dan 4.8.

Tabel 4.8 Data yang belum dinormalisasi

No	Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	Lingkar Kepala	Lingkar Dada	Status
1	1	Laki-laki	Tinggi	Naik	Kecil	Gemuk	Kurang gizi
2	2	Laki-laki	Tinggi	Naik	Kecil	Gemuk	Normal
3	3	Laki-laki	Rendah	Naik	Besar	Gemuk	Normal
4	4	Laki-laki	Rendah	Turun	Besar	Gemuk	Normal
5	5	Laki-laki	Rendah	Turun	Kecil	Kurus	Kurang Gizi
6	1	Perempuan	Rendah	Turun	Besar	Kurus	Kurang Gizi
7	2	Perempuan	Rendah	Turun	Besar	Kurus	Kurang Gizi
8	3	Perempuan	Tinggi	Turun	Besar	Kurus	Normal

9	4	Perempuan	Tinggi	Naik	Kecil	Kurus	Normal
10	5	Perempuan	Tinggi	Naik	Kecil	Gemuk	Kurang Gizi

Dalam tabel 4.8 terdapat contoh 10 data yang belum dinormalisasi dengan variabel *input* Umur, Jenis Kelamin, Tinggi Badan, Berat Badan, Lingkar Kepala, Lingkar Dada yang memiliki nilai umur antara 1 – 5, jenis kelamin laki-laki dan perempuan, tinggi badan tinggi atau rendah, berat badan naik atau turun, lingkar kepala besar atau kecil, lingkar dada gemuk atau kurus, dan statusnya normal atau kurang gizi yang akan dinormalisasikan dengan ketentuan yang akan dibuat sesuai tabel 4.9.

Tabel 4.9 Data yang sudah ternormalisasi

No	Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	Lingkar Kepala	Lingkar Dada	Status
1	0	1	1	1	0	1	-1
2	0.25	1	1	1	0	1	1
3	0.5	1	0	1	1	1	1
4	0.75	1	0	0	1	1	1
5	1	1	0	0	0	0	-1
6	0	0	0	0	1	0	-1
7	0.25	0	0	0	1	0	-1
8	0.5	0	1	0	1	0	1
9	0.75	0	1	1	0	0	1
10	1	0	1	1	0	1	-1

Dari tabel 4.8 data akan dinormalisasikan seperti tabel 4.9 dengan

Keterangan :

1. Umur sesuai nilai normalisasi.
2. Jenis Kelamin (Laki-laki = 1, Perempuan = 0).
3. Tinggi Badan (Tinggi = 1, Rendah = 0).
4. Berat Badan (Naik = 1, Turun = 0).
5. Lingkar Kepala (Besar = 1, Kecil = 0).
6. Lingkar Dada (Gemuk = 1, Kurus = 0).
7. Status ( Normal = 1, Kurang Gizi = -1) .

#### 4.2.4. Proses Pelatihan (training)

Pada tahap ini menggunakan fungsi *kernel*, *konstanta*, dan *max iterasi* untuk membentuk matrik pelatihan berdasarkan pada tabel 4.9 yang digunakan untuk mencari *support vector*.

$$\begin{array}{l}
 X_1 \quad [ \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad ] \\
 X_2 \quad [ \quad 0.25 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad ] \\
 X_3 \quad [ \quad 0.5 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad ] \\
 X_4 \quad [ \quad 0.75 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad ] \\
 X_5 \quad [ \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad ] \\
 X_6 \quad [ \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad ] \\
 X_7 \quad [ \quad 0.25 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad ] \\
 X_8 \quad [ \quad 0.5 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad ] \\
 X_9 \quad [ \quad 0.75 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad ] \\
 X_{10} \quad [ \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad ]
 \end{array}$$

Dengan menggunakan data pada matrik diatas dihitung *dot-product* semua data seperti contoh sebagai berikut :

$$\begin{aligned}
 X_1 X_1 &= [ 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 ] [ 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 ] \\
 &= 4
 \end{aligned}$$

Perhitungan diatas dilakukan untuk semua data. Setelah menghitung perkalian antar data seperti diatas kemudian menghitung fungsi polinomialnya dengan nilai konstanta  $(c) = 1$  dan nilai polinomialnya derajat  $(d) = 2$ .

$$\begin{aligned}
 K(1,1) &= (X_1 X_1 + c)^d \\
 K(1,1) &= (4 + 1)^2 \\
 K(1,1) &= (5)^2 \\
 K(1,1) &= 25
 \end{aligned}$$

$$\begin{aligned}
 X_1 X_2 &= [ 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 ] [ 0.25 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 ] \\
 &= 4
 \end{aligned}$$

$$\begin{aligned}
 K(1,1) &= (X_1 X_1 + c)^d \\
 K(1,1) &= (4 + 1)^2
 \end{aligned}$$

$$K(1,1) = (5)^2$$

$$K(1,1) = 25$$

Perhitungan tersebut dilakukan hingga  $K(10,10)$ . Adapun hasil lengkapnya dapat dilihat pada tabel 4.10.

Tabel 4.10 Fungsi Kernal 10 data

	1	2	3	4	5	6	7	8	9	10
1	25	25	16	9	4	1	1	4	9	16
2	25	25.62891	17.01563	10.16016	5.0625	1	1.128906	4.515625	10.16016	18.0625
3	16	17.01563	27.5625	19.14063	6.25	4	4.515625	5.0625	5.640625	12.25
4	9	10.16016	19.14063	20.81641	7.5625	4	4.785156	5.640625	2.441406	7.5625
5	4	5.0625	6.25	7.5625	9	1	1.5625	2.25	3.0625	4
6	1	1	4	4	1	4	4	4	1	1
7	1	1.128906	4.515625	4.785156	1.5625	4	4.253906	4.515625	1.410156	1.5625
8	4	4.515625	5.0625	5.640625	2.25	4	4.515625	10.5625	5.640625	6.25
9	9	10.16016	5.640625	2.441406	3.0625	1	1.410156	5.640625	12.69141	14.0625
10	16	18.0625	12.25	7.5625	4	1	1.5625	6.25	14.0625	25

Tabel 4.10 adalah hasil perhitungan dari  $K(1,1)$  sampai dengan  $K(10,10)$ . Setelah parameter terhitung dan terpenuhi maka *dibentuk matrik hessian* dengan parameter  $\lambda=1$ . Contoh perhitungan *matrik hessian* didapatkan dari rumus 3.33 pada bab III dengan rumus sebagai berikut  $((Y_i \cdot Y_j) \cdot (K_{1,1} + \lambda^2))$ . Adapun contoh menghitungnya untuk data  $X_{1,1}$  sampai  $X_{10,10}$  adalah sebagai berikut :

$$X(1,1) = ((Y_i \cdot Y_j) \cdot (K_{1,1} + \lambda^2))$$

$$X(1,1) = ((-1 \cdot -1) \cdot (25+1^2))$$

$$X(1,1) = (1 \cdot 26)$$

$$X(1,1) = 26$$

$$X(1,2) = ((Y_i \cdot Y_j) \cdot (K_{1,2} + \lambda^2))$$

$$X(1,2) = ((-1 \cdot 1) \cdot (25+1^2))$$

$$X(1,2) = (-1 \cdot 26)$$

$$X(1,2) = -26$$



Setelah dilakukan perhitungan matrik hessian dari  $X_{1,1}$  sampai  $X_{10,10}$  dengan rumus yang sama didapatkan hasil seperti tabel 4.11.

Tabel 4.11 Matriks Hessian

	1	2	3	4	5	6	7	8	9	10
1	26	-26	-17	-10	5	2	2	-5	-10	17
2	-26	26.62891	18.01563	11.16016	-6.0625	-2	-2.12891	5.515625	11.16016	-19.0625
3	-17	18.01563	28.5625	20.14063	-7.25	-5	-5.51563	6.0625	6.640625	-13.25
4	-10	11.16016	20.14063	21.81641	-8.5625	-5	-5.78516	6.640625	3.441406	-8.5625
5	5	-6.0625	-7.25	-8.5625	10	2	2.5625	-3.25	-4.0625	5
6	2	-2	-5	-5	2	5	5	-5	-2	2
7	2	-2.12891	-5.51563	-5.78516	2.5625	5	5.253906	-5.51563	-2.41016	2.5625
8	-5	5.515625	6.0625	6.640625	-3.25	-5	-5.51563	11.5625	6.640625	-7.25
9	-10	11.16016	6.640625	3.441406	-4.0625	-2	-2.41016	6.640625	13.69141	-15.0625
10	17	-19.0625	-13.25	-8.5625	5	2	2.5625	-7.25	-15.0625	26

Tabel 4.11 yaitu tabel hasil seperti dari perhitungan  $X(1,1)$  sampai  $X(10,10)$ , pada Langkah selanjutnya adalah menghitung  $E_i$ ,  $\delta\alpha_i$ ,  $\alpha_i$  dengan parameter  $\gamma = 0.1$ ,  $C=1$ , *iterasi max* = 5.

$$E_i = \sum_{j=1}^N \alpha_j D_{ij}$$

$$E_i = \sum_{j=1}^{10} \alpha_j D_{ij}$$

$$E_i = (\alpha_1 * D_{11}) + (\alpha_2 * D_{12}) + (\alpha_3 * D_{13}) + (\alpha_4 * D_{14}) + (\alpha_5 * D_{15}) + (\alpha_6 * D_{16}) + (\alpha_7 * D_{17}) + (\alpha_8 * D_{18}) + (\alpha_9 * D_{19}) + (\alpha_{10} * D_{10})$$

$$E_i = (0*26) + (0*-26) + (0*-17) + (0*-10) + (0*5) + (0*2) + (0*2) + (0*-5) + (0*-10) + (0*17)$$

$$E_i = 0$$

$$\delta\alpha_1 = \min \{ \max [ \gamma ( 1 - E_1 ) , - \alpha_1 ], C - \alpha_1 \}$$

$$\delta\alpha_1 = \min \{ \max [ 0.15 ( 1 - 0 ) , - 0 ], 1 - 0 \}$$

$$\delta\alpha_1 = \min \{ \max [ 0.15 , - 0 ], 1 \}$$

$$\delta\alpha_1 = \min \{ 0.15 , 1 \}$$

$$\alpha_1 = \alpha_1 + \delta\alpha_1$$

$$\alpha_1 = 0 + 0.15$$

$$\alpha_1 = 0.15$$

Total perhitungan  $E_i$  dan  $\alpha_i$  dengan nilai 0 terbanyak pada *alpha* baru terdapat dalam perhitungan *iterasi* yang ke- 6, tabel data dapat dilihat pada tabel 4.12.

tabel 4.12 perhitungan  $E_i$  dan  $\alpha_i$  iterasi 6

	Data	$E_i$	$\Delta\alpha$	Max	$\Delta\alpha_i$	alpha baru
Iterasi 6	Data 1	14.34926	-2.00239	-1	-1	0
	Data 2	12.91178	-1.78677	0	0	0
	Data 3	11.50673	-1.57601	0	0	0
	Data 4	9.301922	-1.24529	-1	-1	0
	Data 5	7.667409	-1.00011	-0.76556	-0.76556	0
	Data 6	6.318323	-0.79775	-0.79775	-0.79775	0.089171
	Data 7	2.125911	-0.16889	-0.16889	-0.16889	0.709054
	Data 8	3.183908	-0.32759	-0.14355	-0.14355	0
	Data 9	10.22706	-1.38406	-0.98708	-0.98708	0
	Data 10	4.412135	-0.51182	-0.09296	-0.09296	0

Tabel 4.12 merupakan hasil dari perhitungan *iterasi* sampai ke-6 kalinya. Setelah *iterasi* mencapai *alpha* baru yang *maximal* didapatkan data yang dijadikan data *support vector* yaitu data 6 dan data 7. Setelah diketahui data yang dijadikan *support vector* langkah berikutnya mencari nilai  $w$ . Setelah nilai  $w$  terpenuhi digunakan untuk mencari nilai  $b$  (bias) dengan mengambil satu dari masing-masing data yang berbeda kelas. Didapatkan nilai  $b$  sebesar -1.8781. Jika data sudah terpenuhi maka model SVM siap digunakan untuk klasifikasi.

#### 4.2.5. Proses Testing

Pada proses *testing*, langkah awal yang dilakukan adalah melakukan pemodelan pada data *training* terlebih dahulu untuk dijadikan kunci pada proses klasifikasi, selanjutnya data *testing* akan dimasukkan ke dalam model SVM yang sudah dibuat untuk mengetahui hasil dari data yang telah

dimasukkan. hasil keluaran dari proses *testing* adalah hasil prediksi kelas yang didapat. Proses data yang akan dijadikan *testing* terdapat pada tabel 4.13 sampai 4.16.

Tabel 4.13 Data Testing

Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	Lingkar Kepala	Lingkar Dada	Status
2	Laki-laki	Tinggi	Turun	Kecil	Kurus	Kurang Gizi
3	Perempuan	Rendah	Naik	Besar	Gemuk	Normal

Tabel 4.13 adalah data *testing* akan masih akan dinormalisasi terlebih dahulu. Data *testing* yang sudah dinormalisasi dapat dilihat seperti tabel 4.14.

Tabel 4.14 Data Testing ternormalisasi

Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	Lingkar Kepala	Lingkar Dada	Status
0.25	1	1	0	0	0	-1
0.5	0	0	1	1	1	1

Pada tabel 4.14 memperlihatkan data yang sudah dinormalisasi dan selanjutnya akan dilakukan pemodelan data latih, data yang terpilih menjadi data *support vector* terhadap data *testing* diambil dari *alpha* baru yang sudah didapat yaitu data 6 dan data 7 dari pencarian 10 data diatas (tabel 4.12).

Tabel 4.15 Pemodelan Klasifikasi

Umur	Jenis Kelamin	Tinggi Badan	Berat Badan	Lingkar Kepala	Lingkar Dada	Alpha Baru
0	1	1	0	0	0	0.089171
0.25	0	0	1	1	1	0.709054

Tabel 4.15 di dapatkan dari hasil *iterasi* yang ke-6 yang *alpha* barunya masih memiliki nilai. Data yang sudah didapat selanjutnya akan dihitung model klasifikasinya terlebih dahulu seperti pada contoh perhitungan berikut ini.

Model Klasifikasi.

Diambil dari tabel 4.12, tetapi nilai *alpha* yang didapat tidak

dimasukkan dala proses pemodelan.

$$X_6 = [ 0.5 \ 1 \ 1 \ 0 \ 0 \ 0 ]$$

$$X_7 = [ 1 \ 0 \ 0 \ 1 \ 1 \ 1 ]$$

Data *Testing*.

Diambil dari tabel 4.14 dengan tidak memasukkan nilai statusnya.

$$X_{t1} = [ 0.25 \ 1 \ 1 \ 0 \ 0 \ 0 ]$$

$$X_{t2} = [ 0.5 \ 0 \ 0 \ 1 \ 1 \ 1 ]$$

Adapun contoh perhitungan untuk menghitung  $(X_6, X_{t1})$  adalah sebagai berikut :

$$\begin{aligned} (X_6, X_{t1}) &= [ 0.5 \ 1 \ 1 \ 0 \ 0 \ 0 ] [ 0.25 \ 1 \ 1 \ 0 \ 0 \ 0 ] \\ &= 2.125 \end{aligned}$$

Untuk Menghitung  $K ( X_6, X_{t1})$  sebagai berikut :

$$K (X_6, X_{t1}) = (( X_6, X_{t1} ) + 1 )^2$$

$$K (X_6, X_{t1}) = ( 2.125 + 1 )^2$$

$$K (X_6, X_{t1}) = (3.125)^2$$

$$K (X_6, X_{t1}) = 9.766$$

Untuk menghitung  $\alpha_1 y_1 K( x_6, x_{t1} )$  adalah sebagai berikut :

$$\begin{aligned} \alpha_1 y_1 K(x_6, x_{t1}) &= 0.089171 * 1 * 9.766 \\ &= 0.8708 \end{aligned}$$

Setelah menghitung semua nilai  $\alpha_i y_i K(x_1, x_i)$  terhadap nilai data *testing* pertama langkah selanjutnya yaitu mencari hasil pemodelan. Hasil pemodelan merupakan jumlah dari hasil data yang sudah dimasukkan ke dalam model SVM.

$$\begin{aligned} \Sigma \alpha_i y_i K(x_1, x_i) &= 0.8708 + 0.0892 + 0.7091 \\ &= 1.6691 \end{aligned}$$

Fungsi klasifikasi dihitung dari  $\Sigma \alpha_i y_i K(x_1, x_i)$  yang akan dijumlahkan dengan  $b = -1.8781$  maka didapatkan Hasil sebagai berikut :

$$f(\Phi(X_t)) = 1.6691 + ( -1.8781)$$

$$f(\Phi(X_t)) = -0.209$$

Hasil prediksi kelas didapat dari *sign* dari hasil fungsi klasifikasi

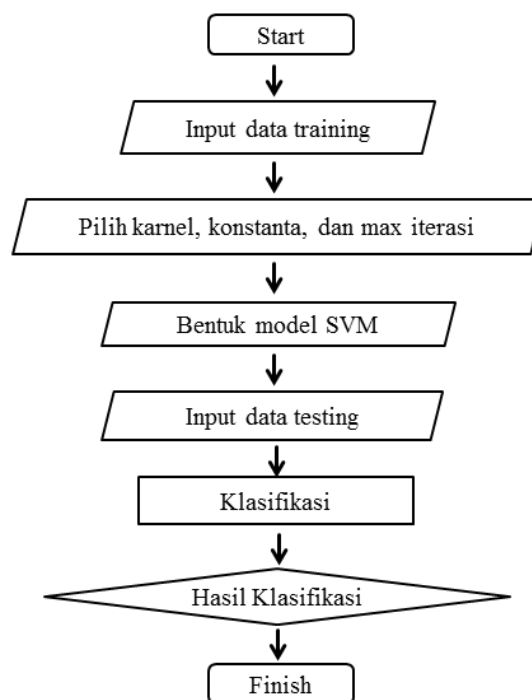
apabila hasilnya lebih dari ( $>$ ) 0 maka masuk kelas 1 dan apabila hasilnya kurang dari ( $<$ ) 0 maka masuk kelas -1. Untuk data *testing* di atas masuk ke dalam kelas -1 yang artinya masuk ke dalam kategori kurang gizi. *Sign* (-0.209) = -1 karena hasilnya -1 maka untuk data *testing* yang pertama masuk ke kategori kurang gizi. Untuk hasil data *testing* lainnya dapat dilihat pada tabel 4.16.

Tabel 4.16 Hasil Klasifikasi Seluruh Data *Testing*

Id data	hasil Prediksi	kelas Asli
1	-1	-1
2	1	1

Pada Tabel 4.16 klasifikasi yang dilakukan pada semua data mendapatkan hasil prediksi -1 dan 1, dan kelas yang didapat menghasilkan kelas asli atau sesuai yang telah dibuktikan dengan langkah-langkah perhitungan diatas.

#### 4.2.6. Flowchat Sistem Klasifikasi Status Gizi Balita



Gambar 4.1 Flowchat Sistem Klasifikasi Status Gizi Balita

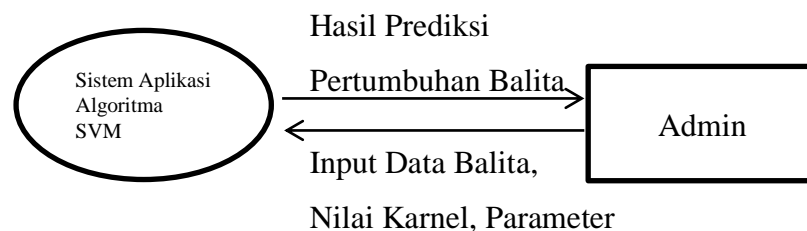
Dijelaskan pada gambar 4.1 Flowchat Sistem Klasifikasi Status Gizi Balita yang diawali dengan *start* (Mulai) dan memasukkan data untuk dibuat pemodelan *training*, selanjutnya pilih *kernel*, *konstanta*, dan *max iterasi* yang di butuhkan kemudian bentuk model SVM. lalu masukkan data yang akan dibuat *testing* dan proses dengan perhitungan klasifikasi maka akan muncul hasil dari perhitungan klasifikasi tersebut.

#### 4.2.7. Data Flow Diagram (DFD)

*Data Flow Diagram* (DFD) atau diagram alir data merupakan diagram yang menggunakan notasi untuk metode pengembangan dari sistem yang terstruktur, DFD dapat menggambarkan seluruh proses kegiatan yang terdapat pada sistem dengan logika dan secara jelas.

##### 4.2.7.1. Konteks Diagram

*Context Diagram* merupakan gambaran umum dari sistem, *context diagram* pada sistem proyek akhir ini dapat dilihat pada gambar 4.3 :

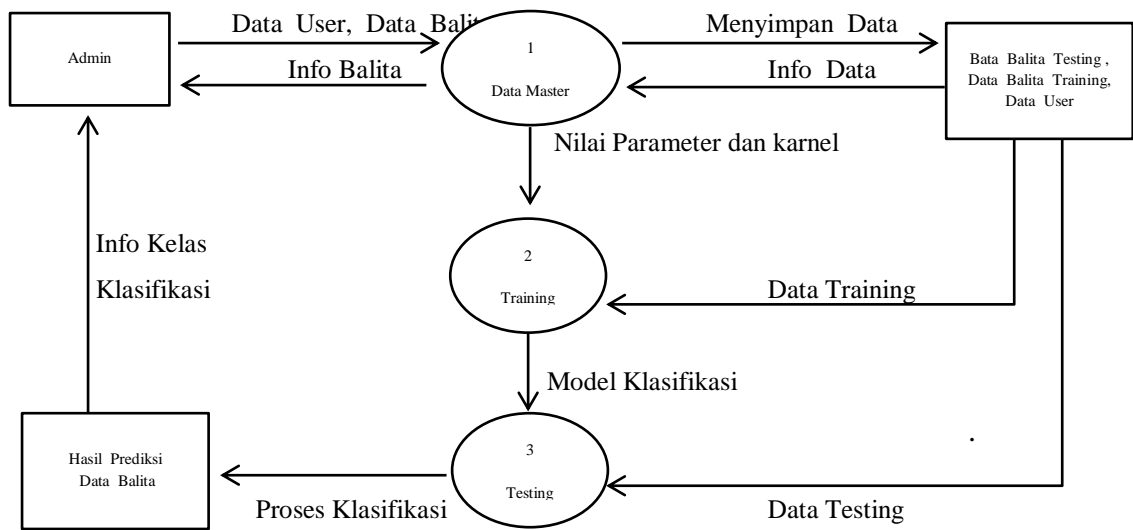


Gambar 4.2 Diagram Konteks

Gambar ini menjelaskan bahwa sistem aplikasi algoritma SVM akan menampilkan hasil prediksi pertumbuhan balita kepada admin ketika admin sudah memasukkan data balita, nilai karnal dan parameter dalam sistem aplikasi algoritma SVM.

##### 4.2.7.2.DFD level 1

DFD level ini merupakan proses selanjutnya dari *context diagram*, dalam level ini data dijelaskan lebih rinci lagi, DFD level 1 dapat di lihat pada gambar 4.3 :



Gambar 4.3 DFD level 1

Dari gambar DFD level 1 (Gambar 4.3) terlihat bahwa sistem memiliki 3 proses yaitu :

a. Master data

Proses ini merupakan proses manajemen data balita yang di butuhkan oleh sistem atau admin, pada proses ini juga terdapat manajemen data berupa menambah, mengedit, menghapus serta menyimpan data balita baik data latih maupun data uji yang nantinya akan di proses dan ditampilkan oleh sistem. Pada proses ini yang menjalankan sistem adalah kader posyandu desa atau pihak yang bertugas dalam suatu posyandu.

b. Training

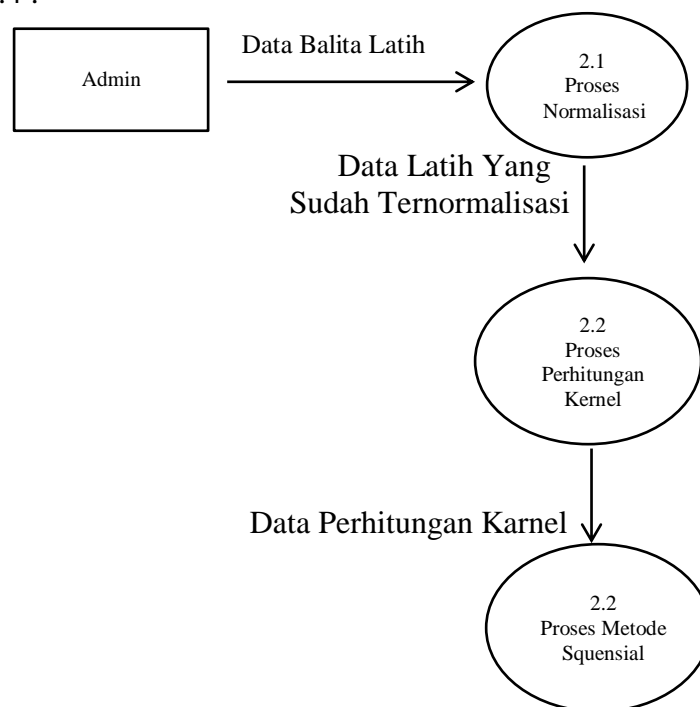
Proses ini adalah awal implementasi SVM pada sistem, dimana pada proses ini setelah sistem menerima data yang sudah diolah oleh master data, selanjutnya akan diterapkan langkah-langkah penyelesaian permasalahannya dengan menggunakan metode sekuensial, proses ini menghasilkan data *support* yang digunakan dalam pemodelan klasifikasi untuk data *testing*. Pada proses *training* orang tua bertindak sebagai *user* yang menentukan berapa jumlah data *training* yang dipakai serta parameter berapa saja yang dipakai.

c. *Testing*

Proses ini merupakan proses pengujian, setelah semua data terpenuhi dan melalui perhitungan dalam proses *training*. Pada tahap akhir adalah proses data *testing* dengan menggunakan model klasifikasi yang sudah dihasilkan pada proses *training* untuk *testing*. *Output* yang dihasilkan merupakan hasil prediksi dari data *input* yang diberikan oleh orang tua untuk di uji. Pada proses ini orang tua dapat mengetahui hasil *output* status gizi balita dari sistem tersebut, apakah masih tetap dalam kondisi baik atau semakin menurun kurang baik.

#### 4.2.7.3. DFD level 2 (Proses Training)

DFD level ini merupakan proses selanjutnya dari DFD level 1, dalam level ini menjelaskan tentang proses *training*, DFD level 2 dapat di lihat pada gambar 4.4 :



Gambar 4.4 DFD level 2 (Proses Training)

Pada DFD Level 2 (gambar 4.4) Terlihat pada proses *training* memiliki tiga proses yaitu :



a. Proses Normalisasi

Merupakan proses untuk menormalisasikan data latih agar dapat diolah. Proses normalisasi penting dikarenakan agar proses dalam perhitungan lebih mudah dan atribut yang rentanya nilainya besar tidak mendominasi atribut yang rentanya lebih kecil.

b. Proses Perhitungan Kernel

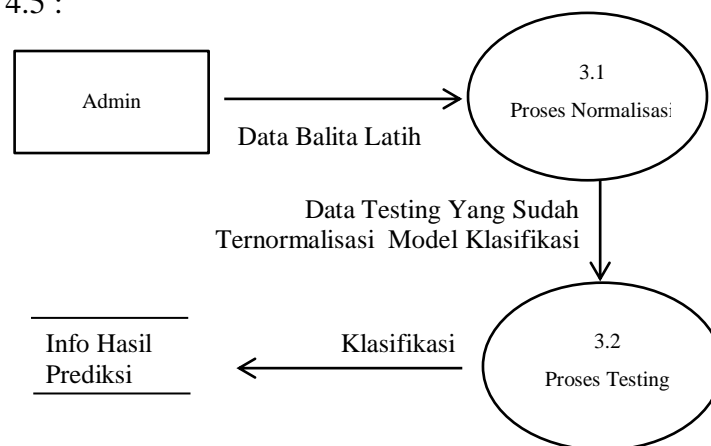
Digunakan untuk membentuk matrik pelatihan yang digunakan untuk mencari *support vector*.

c. Proses Metode sekuensial

Data yang sudah dimodelkan dalam matriks kernel digunakan dalam melakukan pencarian nilai *alpha*. Nilai *alpha* dicari menggunakan metode sekuensial.

#### 4.2.7.4. DFD level 2 (Proes Testing)

DFD level ini merupakan proses selanjutnya dari DFD level 1, dalam level ini menjelaskan tentang proses *testing*. DFD level 2 dapat di lihat pada gambar 4.5 :



Gambar 4.5 DFD level 2 (Proses Testing)

Pada DFD Level 2 (gambar 4.5). Dalam tahap ini memiliki dua proses yaitu :

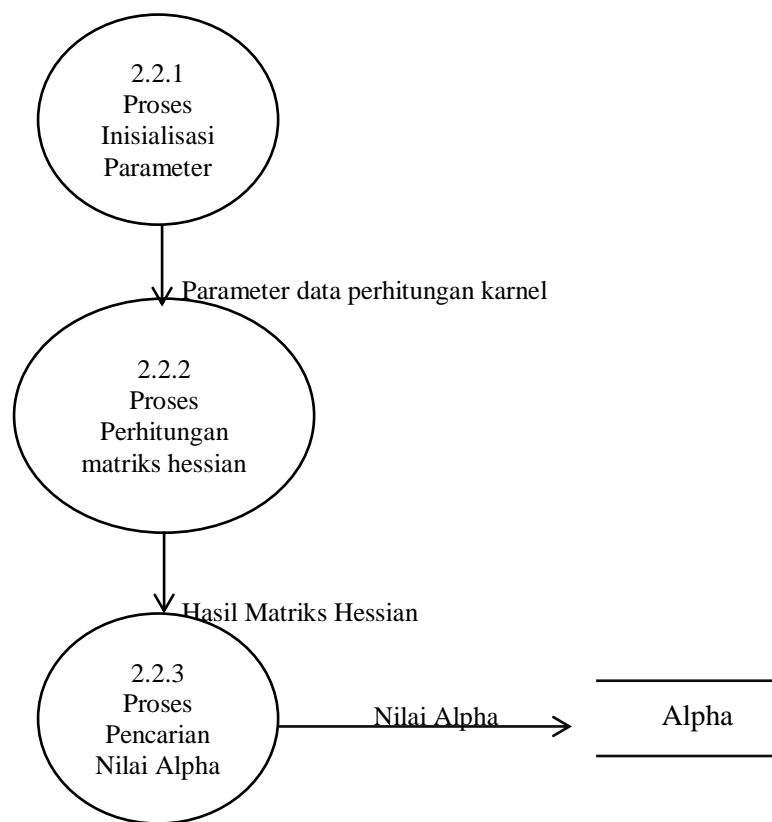
a. Proses Normalisasi

Mempermudah proses perhitungan dan atribut yang rentanya nilainya besar tidak mendominasi atribut yang rentanya lebih kecil.

b. Proses testing

Pada proses ini data *testing* yang sudah dinormalisasi dihitung dengan menggunakan model klasifikasi dari proses hasil *training*. Kemudian hasil dari hasil prediksi ditampilkan Laju *error*, Laju akurasi serta Parameter yang digunakan akan disimpan dalam tabel klasifikasi.

#### 4.2.7.5. DFD Level 3 (Proses Metode Sekuensial)



Gambar 4.6 DFD Level 3 (Proses Metode Sekuensial)

Pada gambar 4.6 terlihat pada proses metode sekuensial memiliki tiga proses yaitu:

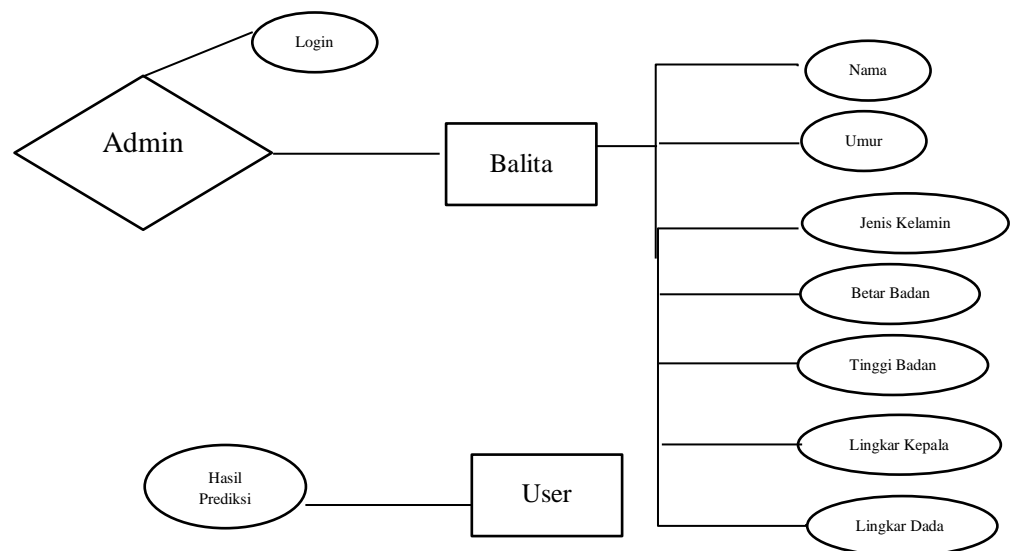
a. Proses inialisasi parameter

merupakan proses untuk menentukan parameter yang dipakai dalam membentuk *matriks hessian*. Parameter tersebut antara lain *Iterasi maximal*,  $\lambda$  (*augmenting factor*),  $\gamma$  (*konstanta learning rate*) dan C sebagai pembatas nilai alpha.

- b. Proses Perhitungan matriks hessian  
proses pembentukan untuk *matriks hessian* berdasarkan parameter yang sudah diinisialisasi dan data *kernel* yang didapatkan.
- c. Proses Pencarian nilai  $\alpha$   
Pencarian nilai *alpha* ini berguna untuk mendapatkan data *training* yang akan digunakan sebagai *support vector*.

#### 4.2.8. Entity Relationship Diagrams (ERD)

**ERD** (*Entity Relationship Diagram*) adalah pemodelan data utama yang membantu mengorganisasikan data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas.



Gambar 4.8 *Entity Relationship Diagrams*

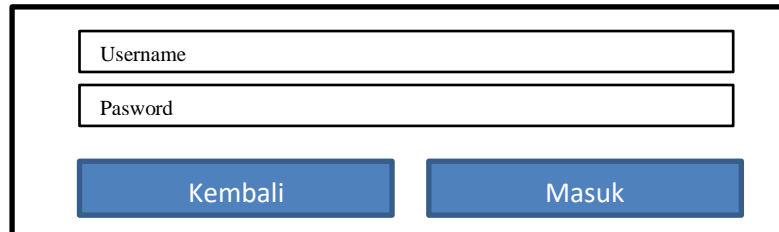
Untuk lebih detail penjelasan ERD Gambar 4.8 akan dijelaskan di bawah ini :

1. Tabel Admin : Berisi data *login* admin ( username, password )
2. Tabel Balita : Berisi detail data balita ( nama, jenis kelamin, umur, berat badan, tinggi badan, lingkar kepala, lingkar dada )
3. Tabel *User* : Sebagai orang tua balita yang hanya bias melihat hasil prediksi perhitungan)

#### 4.2.9. Rencana Antarmuka

Perancangan *input/output* untuk interaksi *user* dan sistem dilakukan pada tahap ini. Berdasarkan dari analisa sistem yang sudah dilakukan maka tampilan antar muka yang dibuat adalah sebagai berikut :

##### 4.2.9.1 Rancangan Login

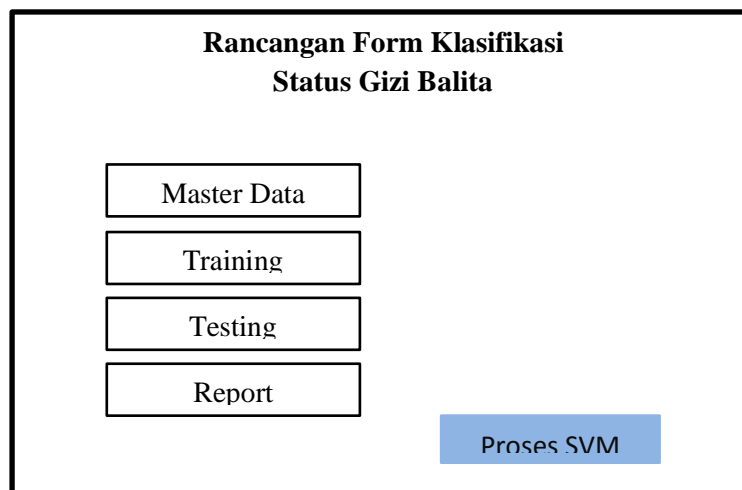


The image shows a login form layout. It consists of two input fields: 'Username' and 'Pasword'. Below these fields are two buttons: 'Kembali' (Back) and 'Masuk' (Login). The buttons are blue with white text.

Gambar 4.8 Rancangan Form Login

Gambar 4.8 adalah rancangan *form login*. *Form* ini berfungsi sebagai jembatan masuk aplikasi *SVM* juga sebagai pengaman aplikasi agar tidak sembarangan digunakan oleh orang yang tidak berkepentingan.

##### 4.2.9.2 Rancangan Form Klasifikasi

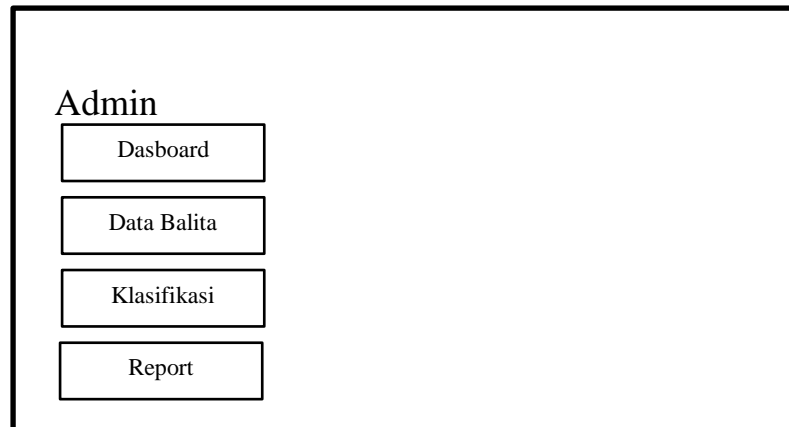


The image shows a classification form layout titled 'Rancangan Form Klasifikasi Status Gizi Balita'. It features four stacked buttons: 'Master Data', 'Training', 'Testing', and 'Report'. To the right of these buttons is a blue button labeled 'Proses SVM'.

Gambar 4.9 Rancangan Form Klasifikasi

Gambar 4.9 adalah rancangan *form* klasifikasi. *Form* ini berfungsi sebagai *form interface* pada aplikasi sistem aplikasi *SVM*. Didalamnya terdapat empat kolom yaitu Master Data (set data yang akan diolah), Training (proses normalisasi, perhitungan karnel dan metode *sequensial*), *Testing* (proses untuk mencapai hasil), dan report (laporan hasil dari capaian).

#### 4.2.9.3 Rancangan Form Admin

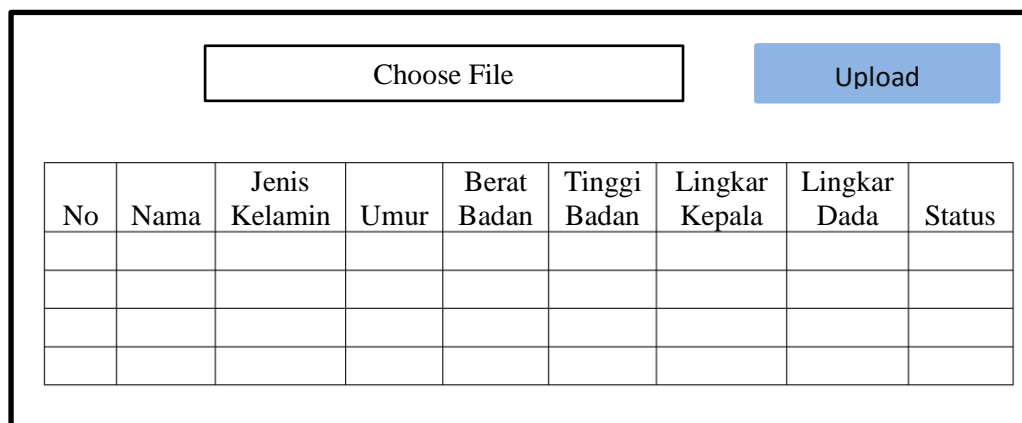


The image shows a rectangular box representing an admin form. At the top left, the word "Admin" is written in a bold, serif font. Below it, there are four rectangular buttons stacked vertically, each containing a menu item: "Dashboard", "Data Balita", "Klasifikasi", and "Report".

Gambar 4.10 Rancangan Form Admin

Gambar 4.10 *Form Admin* adalah *form* yang terbuka setelah hasil *login* yang hanya dapat dibuka oleh admin. Terdapat empat pilihan yaitu *Dashboard* (tampilan utama), *Data Balita* (data keseluruhan yang ada pada database), *Klasifikasi* (proses perhitungan yang digunakan) dan *report* (capaian hasil dari proses perhitungan).

#### 4.2.9.4 Rancangan Form Report



The image shows a rectangular box representing a report form. At the top, there is a "Choose File" button on the left and an "Upload" button on the right. Below these buttons is a table with 9 columns and 4 rows. The columns are labeled: "No", "Nama", "Jenis Kelamin", "Umur", "Berat Badan", "Tinggi Badan", "Lingkar Kepala", "Lingkar Dada", and "Status". The table is currently empty.

No	Nama	Jenis Kelamin	Umur	Berat Badan	Tinggi Badan	Lingkar Kepala	Lingkar Dada	Status

Gambar 4.11 Rancangan Form Report

Gambar 4.11 adalah rancangan *form report* atau *form* yang berfungsi untuk memperoleh hasil laporan dari proses prediksi yang dilakukan, serta hasil perbandingan keakuratan prediksi berdasarkan parameter yang digunakan dalam melakukan proses prediksi status gizi balita. Di sinilah tersimpannya hasil dari semua proses.