BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi merupakan tahap penciptaan perangkat lunak dan juga tahap kelanjutan dari kegiatan perancangan aplikasi. Tahap ini merupakan tahap dimana aplikasi siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, baik itu lingkungan perangkat keras, maupun lingkungan perangkat lunak, serta implementasi program.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk menjalankan *Game* sebagai representasi pengambilan keputusan tidak harus yang berspesifikasi tinggi. Spesifikasi minimum yang dapat digunakan untuk menjalankan *Game* adalah sebagai berikut :

a. Intel Pentium 4

b. RAM 512 MB

c. Tersedia kapasitas minimal sebesar 1 GB

d. VGA Card 256 MB

e. Monitor

f. Mouse

4.1.2 Implementasi Perangkat Lunak

Untuk menjalankan *game* ini diperlukan software khusus yaitu Adobe Flash karena *game* berformat .swf .

Game ini dibangun menggunakan Adobe Flash Professional CS6 dan dengan menggunakan bahasa pemrograman Actionscript 3.0.

4.1.3 Implementasi *Class Diagram*

Adapun implementasi *class diagram* pada *game* yang dibangun adalah sebagai berikut :

1. Kelas MainClass

Implementasi *Class Diagram* pada Kelas MainClass dapat dilihat pada Tabel 4.1.

Tabel 4.1 Implementasi Kelas MainClass

File : MainClass	
Nama Kelas	
Rancangan	**Implementasi**
MainClass	public class MainClass extends MovieClip {}
Atribut	
Rancangan	**Implementasi**
+isMainMenuGame:Boolean +isAboutGame:Boolean +isPlayGame:Boolean +vMainMenuGame:MainMenuGame +vAboutGame:AboutGame +vPlayGame:PlayGame	public var isMainMenuGame, isAboutGame, isPlayGame : Boolean; public var vMainMenuGame : MainMenuGame; public var vAboutGame : AboutGame; public var vPlayGame : PlayGame;
Method	
Rancangan	**Implementasi**
+MainClass() +init():void +showMainMenuGame():void +showAboutGame():void +showPlayGame():void +backToMainGame():void	public function MainClass() {} public function init():void{} public function showMainMenuGame():void{} public function showAboutGame():void{} public function showPlayGame():void{} public function backToMainMenu():void{}

2. Kelas MainMenuGame

Implementasi *Class Diagram* pada Kelas MainMenuGame dapat dilihat pada Tabel 4.2.

Tabel 4.2 Implementasi Kelas MainMenuGame

File : MainMenuGame	
Nama Kelas	
Rancangan	**Implementasi**
MainMenuGame	public class MainMenuGame extends MovieClip {}
Method	
Rancangan	**Implementasi**
+MainMenuGame () +init():void +showPlayGame():void +showAboutGame():void	public function MainMenuGame() {} public function init():void{} public function showPlayGame():void{} public function showAboutGame():void{}

3. Kelas PlayGame

Implementasi *Class Diagram* pada Kelas PlayGame dapat dilihat pada Tabel 4.3.

Tabel 4.3 Implementasi Kelas PlayGame

File : PlayGame	
Nama Kelas	
Rancangan	**Implementasi**
PlayGame	public class PlayGame extends MovieClip {}
Atribut	
Rancangan	**Implementasi**
-objFuzzyLogic:FuzzyLogic	private var objFuzzyLogic : FuzzyLogic;
-objNaiveBayes:NaiveBayes	private var objFuzzyLogic : FuzzyLogic;
-timerLapar:Timer	private var timerLapar:Timer = new Timer(1000, 30);
-timerEnergi:Timer	private var timerEnergi:Timer = new Timer(1000, 30);
-timerKeceriaan:Timer	private var timerKeceriaan:Timer = new Timer(1000, 30);
-sehat:String	private var sehat : String;
-timerTime:Timer	private var timerTime:Timer = new Timer(1000, 60);
-now:Date	private now:Date = new Date();

-hours,minute,seconds:Number +maxHP:int +currentLapar,currentEnergi:int +currentKeceriaan:int +percentLapar,percentEnergi:Number +percentKeceriaan:Number	private hours,minute,seconds:Number; var maxHP:int = 100; var currentLapar, currentEnergi, currentKeceriaan:int; var percentLapar, percentEnergi, percentKeceriaan:Number
Method	
Rancangan	**Implementasi**
+PlayGame() +init():void +setupGame():void +pauseGame():void +lihatProses():void +closeLihatProses():void +inputManual():void +updateBar():void +updateDate():void +showStatusLapar():void +showStatusEnergi():void +showStatusKeceriaan():void +Makan():void +Mandi():void +Tidur():void +Bermain():void +MinumObat():void +hitungFuzzyLogic():void +hitungNaiveBayes():void +timerLaparHandler():void +timerEnergiHandler():void +timerKeceriaanHandler():void	public function PlayGame() {} public function init():void{} public function setupGame():void{} public function showpauseGame():void{} public function lihatProses():void{} public function closeLihatProses():void{} public function inputManual():void{} public function updateBar():void{} public function updateDate():void{} public function showStatusLapar():void{} public function showStatusEnergi():void{} public function showStatusKeceriaan():void{} public function Makan():void{} public function Mandi():void{} public function Tidur():void{} public function Bermain():void{} public function MinumObat():void{} function hitungFuzzyLogic():void{} function NaiveBayes():void{} function timerLaparHandler():void{} function timerEnergiHandler():void{} function timerKeceriaanHandler():void{}

4. Kelas Kucing

Implementasi *Class Diagram* pada Kelas Kucing dapat dilihat pada Tabel 4.4.

Tabel 4.4 Implementasi Kelas Kucing

File : Kucing	
Nama Kelas	
Rancangan	**Implementasi**
Kucing	public class Kucing extends MovieClip {}
Atribut	
Rancangan	**Implementasi**
-nama: String	private var nama:String;
-lapar,energi,keceriaan: int	private var lapar:int; private var energi:int; private var keceriaan:int;
-sehat:int	private var sehat:int;
-param_lapar: String	private var param_lapar:String;
-param_energi: String	private var param_energi:String;
-param_keceriaan: String	private var param_keceriaan:String;
Method	
Rancangan	**Implementasi**
+Kucing()	public function Kucing(){}
+initKucing()	public function initKucing(){}
+setNama():void	public function setNama():void{}
+getNama():String	public function getNama():String{}
+setLapar():void	public function setLapar():void{}
+getLapar():int	public function getLapar():int{}
+setEnergi():void	public function setEnergi():void{}
+getEnergi():int	public function getEnergi():int{}
+setKeceriaan():void	public function setKeceriaan():void{}
+getKeceriaan():int	public function getKeceriaan():int{}
+setSehat():void	public function setSehat():void{}
+getSehat():int	public function getSehat():int{}
+setParamLapar():void	public function setParamLapar():void{}
+getParamLapar():String	public function getParamLapar():String{}
+setParamEnergi():void	public function setParamEnergi():void{}
+getParamEnergi():String	public function getParamEnergi():String{}
+setParamKeceriaan():void	public function setParamKeceriaan():void{}
+getParamKeceriaan():String	public function getParamKeceriaan():String{}
+Makan():void	public function Makan():void{}
+Mandi():void	public function Mandi():void{}
+Bermain():void	public function Bermain():void{}
+Tidur():void	public function Tidur():void{}
+MinumObat():void	public function MinumObat():void{}

5. Kelas FuzzyLogic

Implementasi *Class Diagram* pada Kelas FuzzyLogic dapat dilihat pada Tabel 4.5.

Tabel 4.5 Implementasi Kelas FuzzyLogic

File : FuzzyLogic	
Nama Kelas	
Rancangan	**Implementasi**
FuzzyLogic	public class FuzzyLogic{}
Atribut	
Rancangan	**Implementasi**
-hasil_lapar:Number -kriteria_lapar, kriteria_lapar1, kriteria_lapar2:String -isLapar,isNormalLapar,isKenyang: Boolean -hasil_energi:Number -kriteria_energi, kriteria_energi1, kriteria_energi2:String -isLemah,isNormalEnergi,isKuat: Boolean -hasil_keceriaan:Number -kriteria_keceriaan, kriteria_keceriaan1, kriteria_keceriaan2:String -isSedih, isNormalKeceriaan,isSenang: Boolean -nilai1, nilai2:Number	private var hasil_lapar:Number; private var kriteria_lapar, kriteria_lapar1, kriteria_lapar2:String; private var isLapar, isNormalLapar, isKenyang:Boolean; private var hasil_energi:Number; private var kriteria_energi, kriteria_energi1, kriteria_energi2:String; private var isLemah, isNormalEnergi, isKuat:Boolean; private var hasil_keceriaan:Number; private var kriteria_keceriaan, kriteria_keceriaan1, kriteria_keceriaan2:String; private var isSedih, isNormalKeceriaan, isSenang:Boolean; private var nilai_lapar1,nilai_lapar2:Number; private var nilai_energi1,nilai_energi2:Number; private var nilai_keceriaan1,nilai_keceriaan2:Number;
Method	
Rancangan	**Implementasi**
+FuzzyLogic() +setKriteriaLapar():void +setKriteriaEnergi():void +setKriteriaKeceriaan():void +getKriteriaLapar():String +getKriteriaEnergi():String +getKriteriaKeceriaan():String	public function FuzzyLogic() {} public function setKriteriaLapar():void{} public function setKriteriaEnergi():void{} public function setKriteriaKeceriaan():void{} public function getKriteriaLapar():String{} public function getKriteriaEnergi():String{} public function getKriteriaKeceriaan():String{}

6. Kelas NaiveBayes

Implementasi *Class Diagram* pada Kelas MainClass dapat dilihat pada Tabel 4.6.

Tabel 4.6 Implementasi Kelas NaiveBayes

File : NaiveBayes	
Nama Kelas	
Rancangan	**Implementasi**
NaiveBayes	public class NaiveBayes{}
Atribut	
Rancangan	**Implementasi**
-P_Lapar_Ya,P_Lapar_Tidak:Array	private var P_Lapar_Ya:Array; private var P_Lapar_Tidak:Array;
-P_Energi_Ya,P_Energi_Tidak:Array	private var P_Energi_Ya:Array; private var P_Energi_Tidak:Array;
-P_Keceriaan_Ya, P_Keceriaan_Tidak:Array	private var P_Keceriaan_Ya:Array; private var P_Keceriaan_Tidak:Array;
-P_Ya, P_Tidak:Number	private var P_Ya:Number; private var P_Tidak:Number;
-P1, P2:Number	private var P1, P2:Number;
Method	
Rancangan	**Implementasi**
+NaiveBayes() +init():void +HitungNaiveBayes():String	public function NaiveBayes() {} public function init():void{} public function HitungNaiveBayes():String{}

4.2 Pengujian

Pada tahap ini adalah pengujian terhadap perangkat lunak dan kesesuaian antara hasil perancangan dan tujuan yang diharapkan. Bab ini membahas kelompok proses yang akan diuji dalam sistem rekayasa lalu lintas ini berdasarkan perancangan UML pada bab 3, tujuan pengujian kualitas, menentukan kategori keberhasilan dalam pengujian, perancangan uji kualitas, pelaksanaan pengujian dan kesimpulan berdasarkan hasil pengujian.

4.2.1 Rencana Pengujian

Rencana pengujian akan dilakukan dengan menguji sistem dari segi fungsionalitas. Rencana pengujian perangkat lunak *Game* yang dibangun selengkapnya dapat dilihat pada tabel dibawah 4.7.

Tabel 4.7 Skenario Rencana Pengujian

Item Uji	Detail Pengujian	Jenis Pengujian
viewMain	Klik Tombol *Play*	*Black-Box*
Makan	Aksi Makanan Kucing	*Black-Box*
	Aksi Susu	*Black-Box*
Mandi	Aksi Mandi	*Black-Box*
Bermain	Aksi Bermain Bola	*Black-Box*
	Aksi Bermain *Tickle Stick*	*Black-Box*
Tidur	Aksi Tidur	*Black-Box*
Minum Obat	Aksi Minum Obat	*Black-Box*

4.2.2 Kategori Hasil Pengujian

Dalam menentukan kategori dalam pengujian kualitas pada aplikasi ini dibagi menjadi dua kategori, yaitu :

1. Terpenuhi

Proses yang diuji kualitasnya sesuai dengan tujuan perancangan dan kegunaannya, maka termasuk dalam kategori terpenuhi.

2. Tidak Terpenuhi

Proses yang diuji kualitasnya tidak sesuai dengan tujuan perancangan dan kegunaannya, maka termasuk dalam kategori tidak terpenuhi.

4.2.3 Pelaksanaan Pengujian

Tahap ini yaitu menjalankan perangkat lunak dengan cara menjalankan *game*, pelaksanaan pengujian ini adalah sebagai berikut.

1. Pengujian viewMain

Tabel 4.8 menunjukkan detail pengujian viewMain.

Tabel 4.8 Pengujian viewMain

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button Play*	Masuk ke antarmuka permainan	Antarmuka permainan ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button Play*	Tidak menampilkan antarmuka permainan	Antarmuka permainan tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

2. Pengujian Aksi Makanan Kucing

Tabel 4.9 menunjukkan detail pengujian Aksi Makanan Kucing.

Tabel 4.9 Pengujian Aksi Makanan Kucing

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Makanan Kucing	Karakter pada *Game* menampilkan aksi makan makanan kucing	Aksi makan makanan kucing ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Makanan Kucing	Karakter pada *Game* tidak menampilkan aksi makan makanan kucing	Aksi makan makanan kucing tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

3. Pengujian Aksi Minuman Susu

Tabel 4.10 menunjukkan detail pengujian Aksi Minuman Susu.

Tabel 4.10 Pengujian Aksi Minuman Susu

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Minuman Susu	Karakter pada *Game* menampilkan aksi minum minuman susu	Aksi minum minuman susu ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Minuman Susu	Karakter pada *Game* tidak menampilkan aksi minum minuman susu	Aksi minum minuman susu tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

4. Pengujian Aksi Mandi

Tabel 4.11 menunjukkan detail pengujian Aksi Mandi.

Tabel 4.11 Pengujian Aksi Mandi

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Mandi	Karakter pada *Game* menampilkan aksi mandi	Aksi mandi ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Mandi	Karakter pada *Game* tidak menampilkan aksi mandi	Aksi mandi tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

5. Pengujian Aksi Bermain Bola

Tabel 4.12 menunjukkan detail pengujian Aksi Bermain Bola.

Tabel 4.12 Pengujian Aksi Bermain Bola

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Bola	Karakter pada *Game* menampilkan aksi bermain bola	Aksi bermain bola ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Bola	Karakter pada *Game* tidak menampilkan aksi bermain bola	Aksi bermain bola tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

6. Pengujian Aksi Bermain *Tickle Stick*

Tabel 4.13 menunjukkan detail pengujian Aksi Bermain *Tickle Stick*.

Tabel 4.13 Pengujian Aksi Bermain *Tickle Stick*

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button Tickle Stick*	Karakter pada *Game* menampilkan aksi bermain *tickle stick*	Aksi bermain *tickle stick* ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button Tickle Stick*	Karakter pada *Game* tidak menampilkan aksi bermain *tickle stick*	Aksi bermain *tickle stick* tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

7. Pengujian Aksi Tidur

Tabel 4.14 menunjukkan detail pengujian Aksi Tidur.

Tabel 4.14 Pengujian Aksi Tidur

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Tidur	Karakter pada *Game* menampilkan aksi tidur	Aksi tidur ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Tidur	Karakter pada *Game* tidak menampilkan aksi tidur	Aksi tidur tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

8. Pengujian Aksi Minum Obat

Tabel 4.15 menunjukkan detail pengujian Aksi Minum Obat.

Tabel 4.15 Pengujian Aksi Minum Obat

Kasus dan Hasil Uji (Data Normal)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik kiri *(Mouse* atau *Touchpad)* pada *button* Obat	Karakter pada *Game* menampilkan aksi minum obat	Aksi minum obat ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi
Kasus dan Hasil Uji (Data Salah)			
Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol lain selain klik kiri *(Mouse* atau *Touchpad)* pada *button* Obat	Karakter pada *Game* tidak menampilkan aksi minum obat	Aksi minum obat tidak ditampilkan	[∨]Terpenuhi [ ]Tidak Terpenuhi

4.2.4 Kesimpulan Hasil *Black Box Testing*

Berdasarkan hasil pengujian *black box testing* dari 8 kasus uji fungsional, semuanya berhasil sehingga dapat disimpulkan bahwa implementasi sistem secara fungsional telah sesuai dengan perancangan pada bab 3.

Hasil pengujian dengan menggunakan *black box testing* menunjukkan bahwa seluruh kasus uji menghasilkan nilai terpenuhi yang artinya sistem ini sudah sesuai dengan tujuan awal penelitian dan memenuhi persyaratan sistem yang ditetapkan pada bab 3 analisa dan perancangan kebutuhan sistem.

4.3 Analisis Hasil Uji

Tahap ini akan menganalisis *game* pada saat proses *pre-processing* dengan Logika Fuzzy dan perhitungan Naïve Bayes. Pada sub bab kali ini, analisis hasil uji yang pertama mengenai variabel yang mempengaruhi perubahan keadaan hewan menggunakan *Fuzzy Logic*. Variabel atau faktor-faktor tersebut antara lain adalah Kelaparan, Energi, dan Keceriaan. Selanjutnya akan dilakukan pengujian perhitungan *Naïve Bayes.*

4.3.1 Uji *Pre-Processing* Variabel pada Hewan

Analisis hasil uji meliputi proses perhitungan *pre-proccesing* variabel yang mempengaruhi perubahan keadaan hewan menggunakan *Fuzzy Logic* yang dilakukan secara manual dan proses perhitungan yang dilakukan oleh sistem. Kemudian hasil dari kedua perhitungan tersebut dibandingakan untuk ditarik kesimpulannya.

Data yang digunakan untuk hasil uji ini menggunakan data yang telah dibahas pada bab sebelumnya. Tabel 4.16 merupakan data awal variabel yang mempengaruhi perubahan keadaan hewan.

Tabel 4.16 Data Awal Variabel Hewan

Variabel	Nilai
Kelaparan	55
Energi	65
Keceriaan	40

Proses perhitungan *preprocessing* dilakukan dengan menggunakan logika *fuzzy* yang dalam hal ini dengan menggunakan fungsi keanggotaan *fuzzy*. Data hasil perhitungan manual dapat dilihat pada table 4.17 :

Tabel 4.17 Data Hasil Perhitungan Manual

Variabel	Nilai Bobot Optimal	Kriteria
Lapar	0,8	Normal
Energi	0,75	Normal
Keceriaan	0,2	Normal

Selanjutnya, membandingan hasil perhitungan manual dengan hasil perhitungan yang dilakukan oleh sistem terlihat pada tabel 4.18 :

Tabel 4.18 Perbandingan Hasil Perhitungan Manual dan Program

Variabel	Nilai	Hasil Perhitungan Manual	Hasil Perhitungan Program					Kesimpulan Hasil Uji
			Variabel	Kriteria 1 & 2	Nilai Bobot 1& 2	Kriteria	Nilai Bobot Optimal	
Kelaparan (0 –100)	55	0,8 (Normal)	Lapar 55	Lapar Normal	0 0.8	Normal	0.8	Sesuai
Energi (0 – 100)	65	0,75 (Normal)	Energi 65	Normal Kuat	0.75 0	Normal	0.75	Sesuai
Keceriaan (0 – 100)	40	0,2 (Normal)	Keceriaan 40	Sedih Normal	0 0.2	Normal	0.2	Sesuai

4.3.2 Uji Perhitungan *Naïve Bayes*

Hasil dari proses perhitungan dengan logika fuzzy selanjutnya digunakan sebagai data masukan untuk proses perhitungan dengan menggunakan Naïve Bayes. Dalam uji perhitungan Naïve Bayes ini digunakan juga data sebelumnya yang telah dibahas pada bab lalu. Data masukan Naïve Bayes dapat dilihat pada tabel 4.19.

Tabel 4.19 Data Masukan *Naïve Bayes*

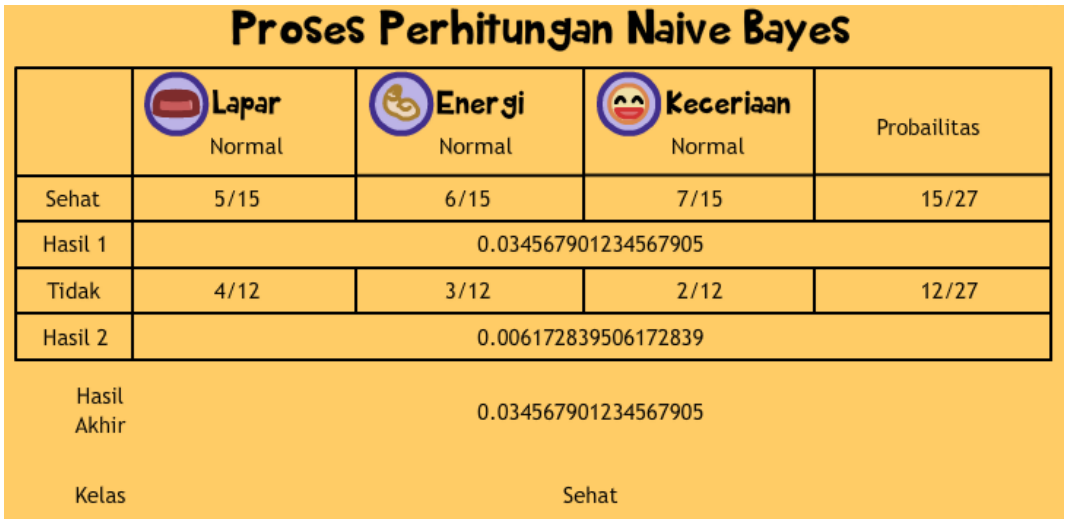
Variabel	**Nilai**	**Masukan untuk *Naïve Bayes***
Kelaparan	55	Normal
Energi	65	Normal
Keceriaan	40	Normal

Setelah data masukan diperoleh maka proses perhitungan menggunakan Naïve Bayes dimulai. Perhitungan secara manual proses Naïve Bayes dapat dilihat pada tabel 4.20.

Tabel 4.20 Hasil Perhitungan *Naïve Bayes*

Kriteria Variabel			**Kelas 1 & 2**	**Nilai Probabilitas**	**Keadaan**
Kelaparan	**Energi**	**Keceriaan**			
Normal	Normal	Normal	Sehat	0,0345	Sehat
			Tidak Sehat	0,0061	

Selanjutnya, membandingan hasil perhitungan manual dengan hasil perhitungan yang dilakukan oleh sistem. Hasil perhitungan *Naïve Bayes* yang dilakukan oleh sistem dapat dilihat pada gambar 4.1 :



Gambar 4.1 Hasil Perhitungan *Naïve Bayes*

4.3.3 Kesimpulan Hasil Uji

Berdasarkan hasil pengujian *pre-processing* yang dilakukan, terlihat hasil perhitungan *Fuzzy Logic* dan *Naïve Bayes* secara manual dengan hasil perhitungan pada program memiliki hasil keluaran yang sama sehingga dapat disimpulkan bahwa implementasi sistem algoritma telah sesuai dengan proses analisis algoritma yang tertera pada bab 3.

Hasil pengujian *pre-processing* menunjukkan bahwa seluruh kasus uji menghasilkan nilai terpenuhi yang artinya sistem ini sudah sesuai dengan tujuan awal penelitian dan memenuhi persyaratan sistem yang ditetapkan pada bab 3.

4.4 *Confusion Matrix*

Confusion Matrix digunakan untuk mengukur akurasi berdasarkan hasil uji terhadap *game* yang dibangun. Pada tabel 4.21 menunjukkan tabel hasil uji akurasi yang akan diolah menggunakan *Confusion Matrix* untuk diukur akurasinya.

Tabel 4.21 Hasil Uji Akurasi

No.	Kelaparan	Energi	Keceriaan	Prediksi	Hasil	Kesesuaian
1	Lapar	Lemah	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
2	Lapar	Lemah	Normal	Tidak Sehat	Tidak Sehat	Sesuai
3	Lapar	Lemah	Senang	Tidak Sehat	Sehat	Tidak sesuai
4	Lapar	Normal	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
5	Lapar	Normal	Normal	Sehat	Sehat	Sesuai
6	Lapar	Normal	Senang	Sehat	Sehat	Sesuai
7	Lapar	Kuat	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
8	Lapar	Kuat	Normal	Sehat	Sehat	Sesuai
9	Lapar	Kuat	Senang	Sehat	Sehat	Sesuai
10	Normal	Lemah	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
11	Normal	Lemah	Normal	Tidak Sehat	Sehat	Tidak sesuai
12	Normal	Lemah	Senang	Sehat	Sehat	Sesuai
13	Normal	Normal	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
14	Normal	Normal	Normal	Sehat	Sehat	Sesuai
15	Normal	Normal	Senang	Sehat	Sehat	Sesuai
16	Normal	Kuat	Sedih	Tidak Sehat	Tidak Sehat	Sesuai

17	Normal	Kuat	Normal	Sehat	Sehat	Sesuai
18	Normal	Kuat	Senang	Sehat	Sehat	Sesuai
19	Kenyang	Lemah	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
20	Kenyang	Lemah	Normal	Sehat	Sehat	Sesuai
21	Kenyang	Lemah	Senang	Sehat	Sehat	Sesuai
22	Kenyang	Normal	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
23	Kenyang	Normal	Normal	Sehat	Sehat	Sesuai
24	Kenyang	Normal	Senang	Sehat	Sehat	Sesuai
25	Kenyang	Kuat	Sedih	Tidak Sehat	Tidak Sehat	Sesuai
26	Kenyang	Kuat	Normal	Sehat	Sehat	Sesuai
27	Kenyang	Kuat	Senang	Sehat	Sehat	Sesuai

Berikut adalah hasil perhitungan akurasi menggunakan *Confusion Matrix* ...(2.5):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{10+15}{10+15+0+2}$$

$$= \frac{25}{27} = 0{,}9295$$

Berdasarkan hasil pengujian akurasi yang dilakukan menggunakan *Confusion Matrix*, didapatkan nilai akurasi dari algoritma yang digunakan adalah 0,9295 atau sebesar 92,95%.