BAB 3
ANALISIS KEBUTUHAN IMPLEMENTASI ALGORITMA

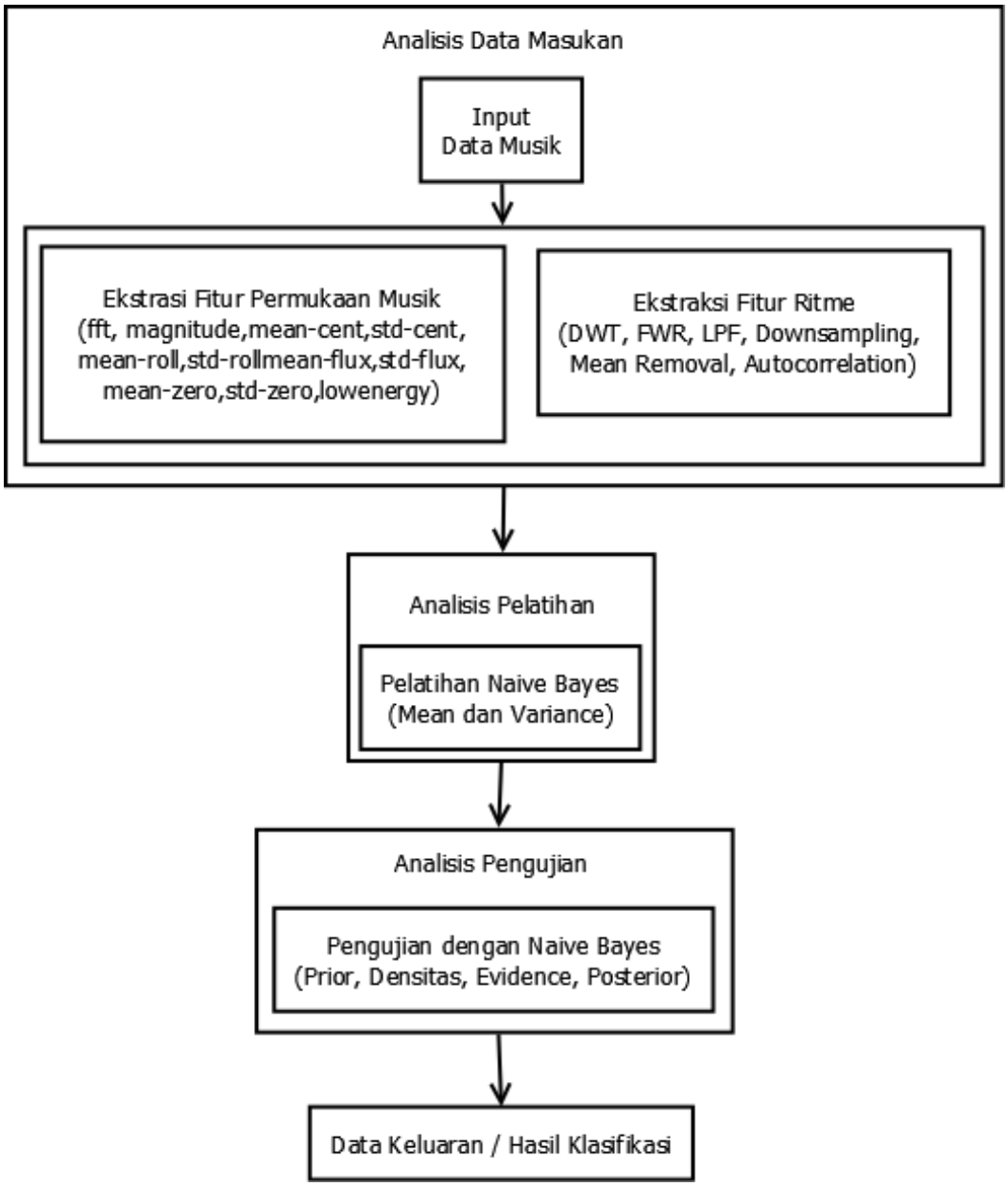
3.1 Analisis Masalah

Dalam mengklasifikasi genre musik, suatu data musik harus diekstraksi untuk mendapatkan ciri atau fitur dari sebuah genre musik tersebut. Hasil ekstraksi yang didapat dari data musik tersebut akan menunjukkan ciri-ciri dari sebuah genre musik yang nantinya akan dihitung dan dibandingkan dengan data musik yang ingin diketahui genrenya. Metode yang digunakan untuk mengklasifikasi genre tersebut dijadikan suatu pertimbangan untuk mendapatkan hasil yang baik. Metode yang akan digunakan pada penelitian kali ini adalah metode fitur permukaan musik dan fitur ritme sebagai ekstraksi fitur dan metode Naive Bayes untuk klasifikasinya.

Sebelumnya telah dilakukan penelitian oleh [1] yang membuktikan bahwa fitur permukaan musik dan fitur ritme dapat mengekstraksi ciri dengan baik. Pada ekstraksi fitur permukaan musik akan dihitung nilai *mean-centroid, mean-rolloff, mean-flux, mean-zerocrossings, std-centroid, std-rollof, std-flux, std-zerocrossings, dan lowenergy* dan pada fitur ritme akan dihitung 4 nilai tertinggi hasil dari perhitungan *autocorrelation.* Ada pula penelitian oleh [2] yang melakukan klasifikasi genre musik dengan *naive bayes classifier* dan menggunakan *Big Endian* dan *Low Endian* dan menghasilkan akurasi 60%. Dari yang telah dijabarkan, maka dalam penelitian kali ini akan melakukan analisis dengan mengkombinasikan fitur permukaan musik dan fitur ritme sebagai ekstraksi fiturnya dengan naive bayes untuk klasifikasi genre musik.

3.2 Analisis Proses

Analisis proses klasifikasi genre musik yang dibangun memiliki 4 tahapan yaitu analisis data masukan, analisis pelatihan, analisis pengujian dan analisis data keluaran seperti yang ditunjukkan pada gambar 3.1.



Gambar 3. 1 Alur Proses

3.2.1 Analisis Data Masukan

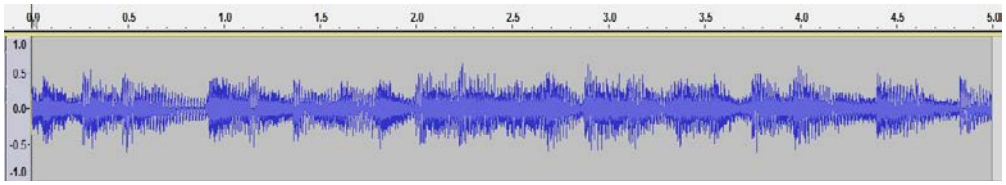
Dalam penelitian ini, yang pertama akan dilakukan adalah analisis data masukan. Analisis data masukan dilakukan untuk mendapatkan nilai inputan yang nantinya dapat digunakan pada proses klasifikasi naive bayes. Data masukan berasal dari data musik yang diambil dari GTZ *dataset collection* yang berjumlah

1000 lagu yang terdiri dari 10 genre dengan setiap genre terdapat 100 lagu didalamnya. Data masukan tersebut berdurasi 30 detik dengan format (.au) seperti

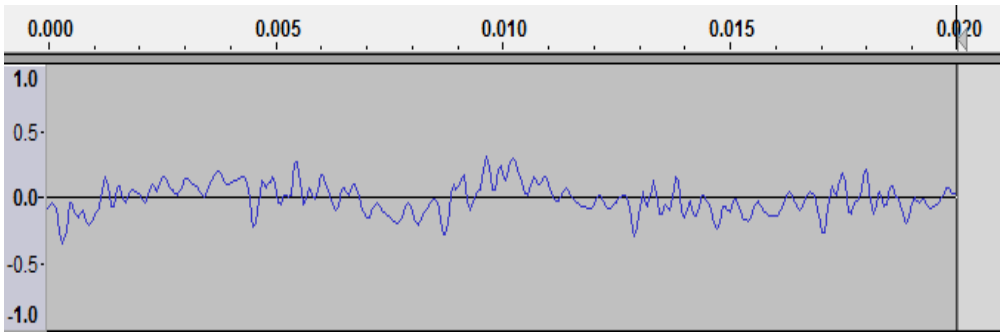
yang ditunjukkan pada gambar 3.2. Untuk memperoleh data masukan yang akan digunakan pada penelitian ini, data musik akan dirubah ke dalam format (.wav) dengan durasi 5 detik seperti yang ditunjukkan pada gambar 3.3. Data masukan tersebut akan dicari nilainya dengan menggunakan ekstraksi fitur permukaan musik dan ekstraksi fitur ritme. Nilai ekstraksi fitur permukaan musik yang akan diperoleh nilainya adalah *mean-centroid, std-centroid, mean-rolloff, std-rolloff ,mean-flux, std-flux, mean-zerocrossings, std-zsserocrossings dan lowenergy*. Sedangkan nilai ekstraksi fitur ritme yang akan dihitung nilainya adalah nilai fwr, lpf, *downsampling, mean-removal* dan *autocorrelation.* Keluaran yang akan didapatkan dari ekstraksi fitur ritme adalah nilai dari 4 titik tertinggi dari proses *autocorrelation.* Nilai-nilai tersebut akan dijadikan sebagai data awal yang akan digunakan sebagai inputan dalam metode naive bayes. Tahapan-tahapan yang akan dilakukan sebelum melakukan ekstraksi fitur permukaan musik adalah melakukan *analysis windowing*, menghitung nilai fft dan nilai *magnitude*. Hasil dari nilai *magnitude* terebut akan menjadi nilai masukan dalam ekstraksi fitur permukaan musik. Sedangkan tahapan perhitungan sebelum melakukan ektraksi fitur ritme adalah menghitung nilai dwt. Hasil dari nilai dwt tersebut akan menjadi masukan nilai masukan dalam ekstraksi fitur ritme.

3.2.1.1 *Analysis Window*

Pada proses *analysis window*, data masukan yang berdurasi 5 detik akan dibagi menjadi 250 *analysis window* yang berdurasi 0.02 detik pada setiap *analysis window*-nya. Hasil dari proses inilah yang nantinya akan dimasukkan dalam proses fft. Ilustrasi dari proses ini dapat dilihat pada gambar 3.2 dan gambar 3.3.



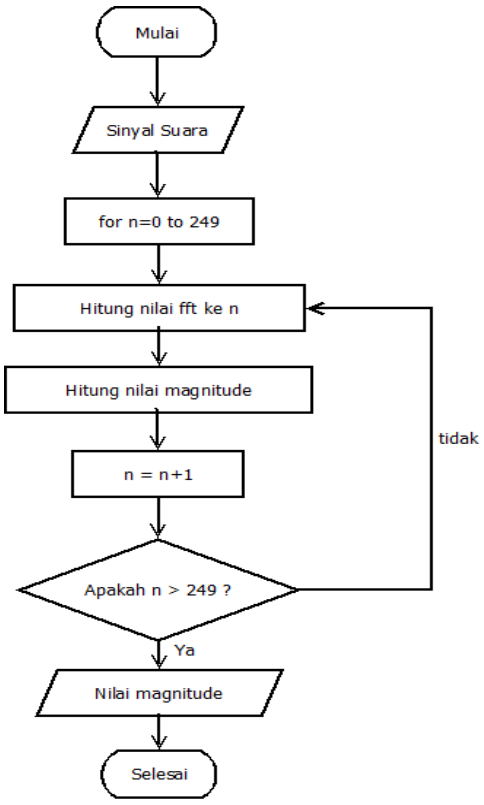
Gambar 3.2 Data Musik 5 Detik



Gambar 3. 3 Data Musik Setelah *Analysis Windowing*

3.2.1.2 *Fast Fourier Transform* (FFT)

Analisa berdasarkan *fourier transform* sama artinya dengan analisa spektrum, karena *fourier transform* merubah signal digital dari domain waktu ke domain frekuensi. Berikut alur proses dari FFT seperti yang ditunjukkan pada gambar 3.4.



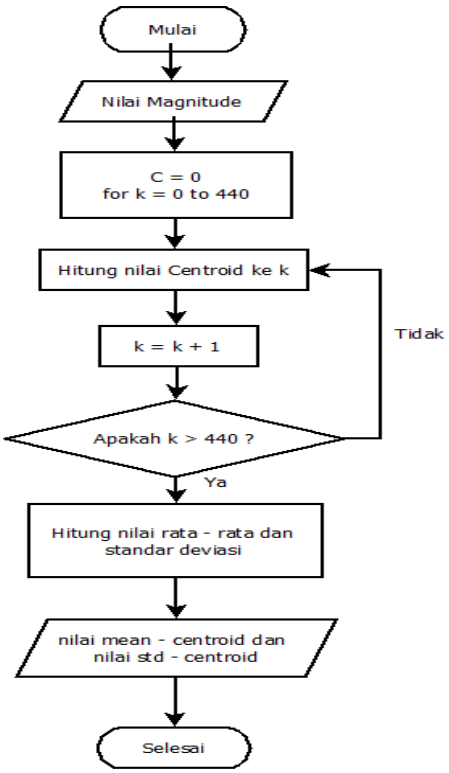
Gambar 3. 4 Alur Proses FFT

3.2.1.3 Ekstraksi Fitur Permukaan Musik

Ekstraksi fitur permukaan musik merupakan proses untuk mendapatkan ciri atau fitur yang terdapat pada musik. Data musik yang telah dihitung dengan fft dan didapatkan nilai *magnitude-nya* akan digunakan pada tahap ini, Tahapan ini akan menghitung nilai *centroid, rolloff, flux, zerocrossings* dan *low energy.* Dari perhitungan tersebut akan diambil 9 nilai keluaran yang akan digunakan sebagai masukan pada proses selanjutnya yaitu *mean-centroid, std-centroid, mean-rolloff, std-rolloff, mean-flux, std-flux, mean-zerocrossings, std-zerocrossings*, dan *low energy.*

1. *Spectral Centroid*

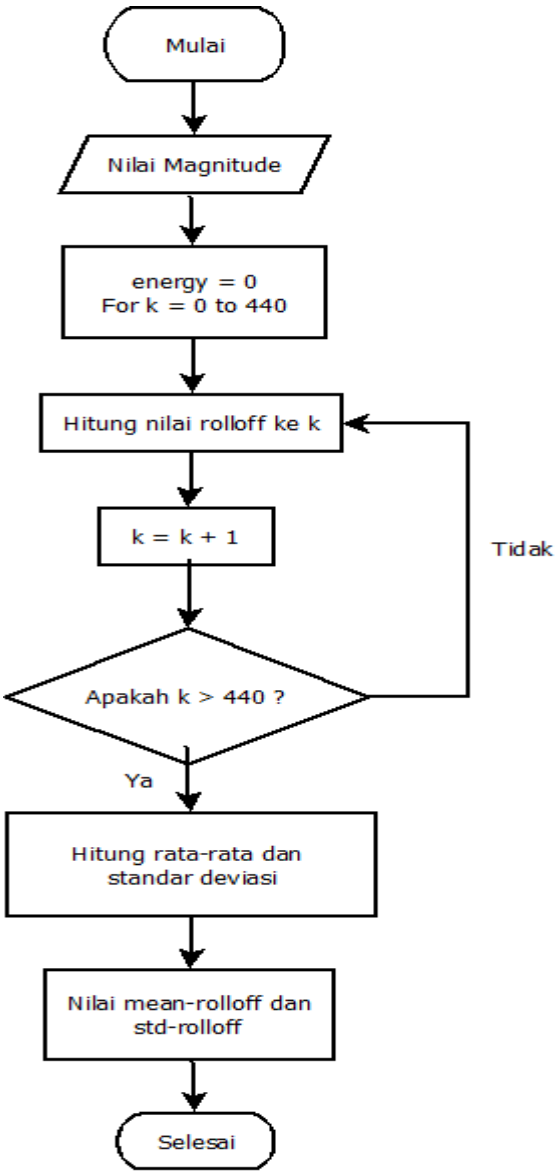
Tahapan pertama dari fitur permukaan musik adalah *spectral centroid. Spectral centroid* menunjukkan pusat grafitasi dari sebuah spektrum. *Centroid* berkorelasi dengan *pitch dan brightness.* Suara dengan *pitch* yang tinggi akan mempunyai nilai *centroid* yang tinggi. Berikut alur proses dari *spectral centroid* seperti yang ditunjukkan pada gambar 3.5.



Gambar 3. 5 Alur Proses *Spectral Centroid*

2. *Spectral Rolloff*

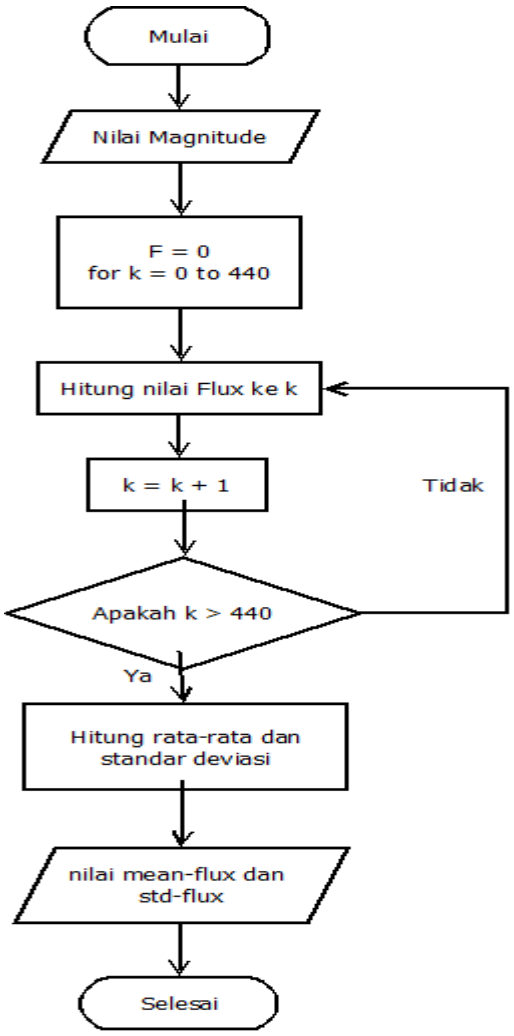
Tahapan kedua dari fitur permukaan musik adalah *spectral rolloff*. *Spectral rolloff* merupakan frekuensi dibawah 85% dari energi pada *magnitude spectrum* yang terkonsentrasi. *Spectral rolloff* merupakan keseimbangan distribusi energy diantara frekuensi tinggi dan rendah pada spektrum. Berikut alur proses dari *spectral rolloff* seperti yang ditunjukkan pada gambar 3.6.



Gambar 3. 6 Alur Proses *Spectral Rolloff*

3. *Spectral Flux*

Tahapan ketiga dari fitur permukaan musik adalah *Spectral Flux. Spectral Flux* adalah ukuran dari *spectral change. Flux* menunjukkan berapa banyak perubahan spektral dari waktu ke waktu. Berikut alur proses dari *spectral flux* seperti yang ditunjukkan pada gambar 3.7.

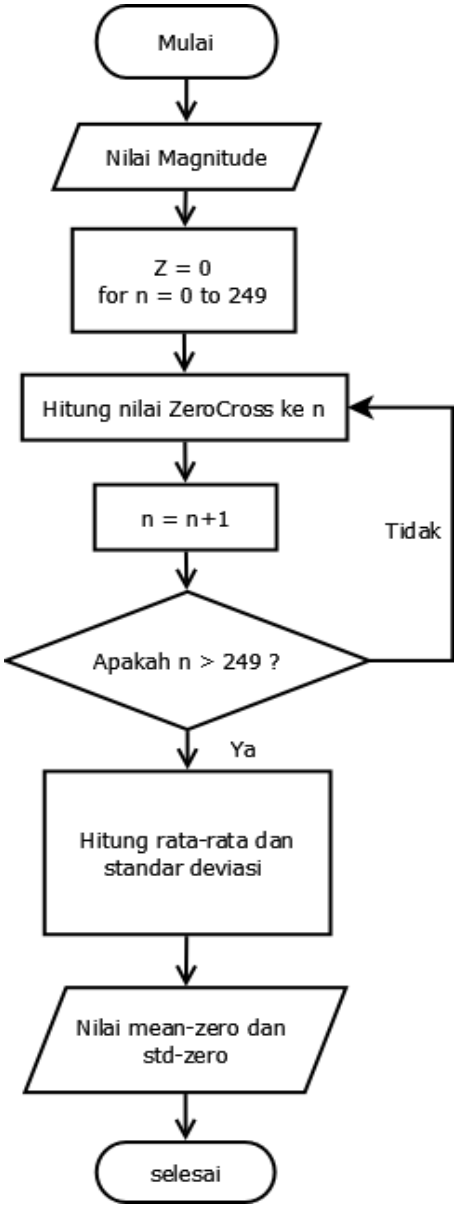


Gambar 3. 7 Alur Proses *Spectral Flux*

4. *Zerocrossings*

Tahapan keempat pada fitur permukaan musik adalah *zerocrossings. Zerocrossings* adalah jumlah dari penyeberangan sinyal dalam 1 *window* yang melalui nilai 0. Pada tahapan ini akan dihitung berapa banyak sinyal melintasi

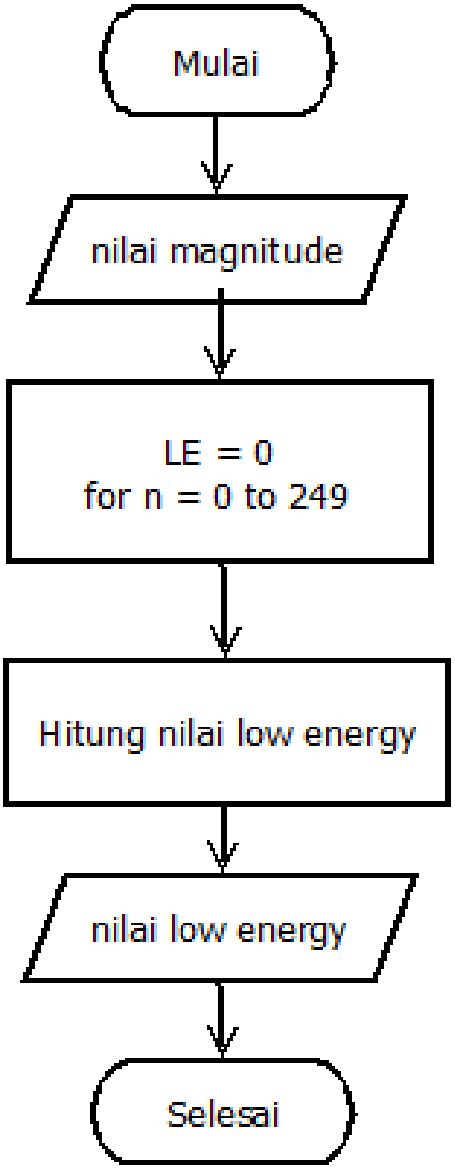
nilai 0 dalam 1 *analysis window*. Berikut alur proses dari *zerocrossings* seperti yang ditunjukkan pada gambar 3.8.



Gambar 3. 8 Alur Proses Zerocrossings

5. *Low Energy*

LowEnergy adalah persentase dari jumlah *analysis windows* yang memiliki energi kurang dari rataan seluruh *analysis windows* dalam satu data musik. Berikut alur proses dari *low energy* seperti yang ditunjukkan pada gambar 3.9.

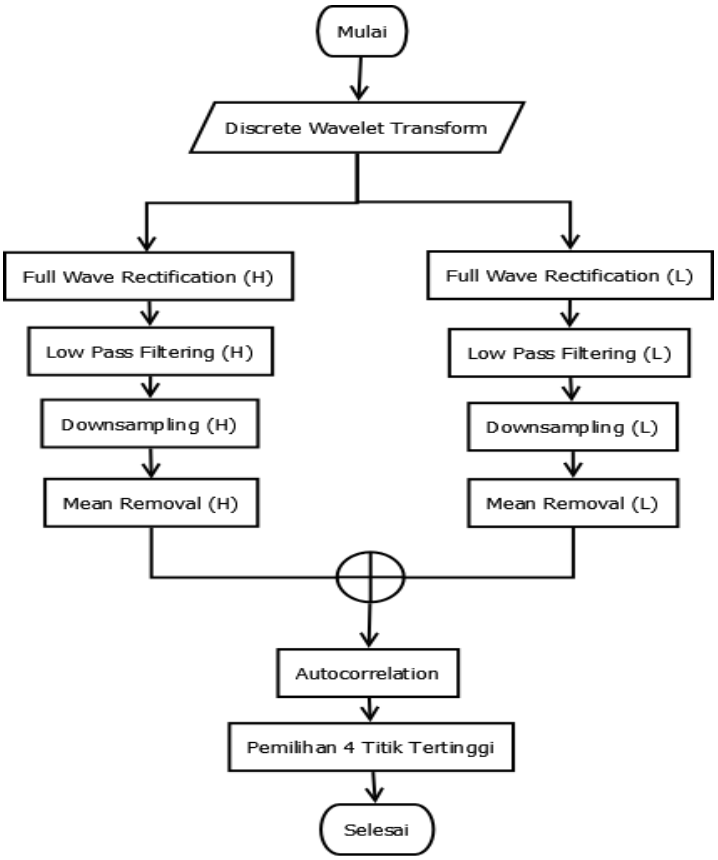


Gambar 3. 9 Alur Proses *Low Energy*

3.2.1.4 Ekstraksi Fitur Ritme

Fitur ritme dikarakterisasikan sebagai pergerakan sinyal musik terhadap waktu dan mengandung informasi seperti yang biasanya terdapat pada ritme, *beat, tempo* dan *time signature*. Fitur yang diset untuk merepresentasikan struktur ritme berasal dari mendeteksi periode yang paling penting pada sebuah sinyal. Ekstraksi fitur ritme merupakan proses untuk mendapatkan ciri atau fitur yang terdapat pada musik., Tahapan ini akan menghitung nilai *dwt, full wave rectification, low pass*

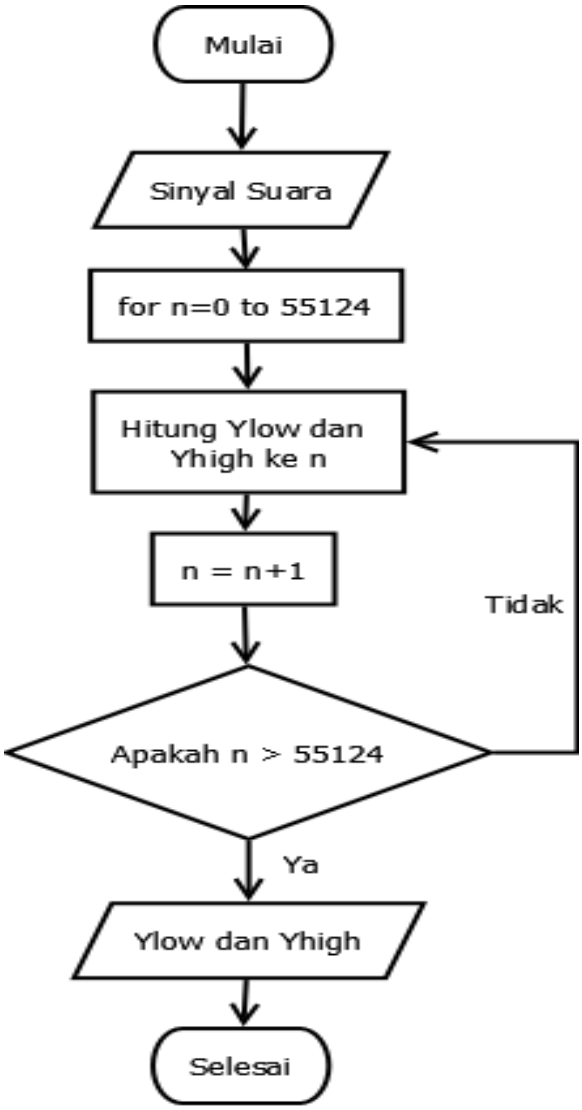
filtering, downsampling, mean removal dan *autocorrelation.* Dari perhitungan tersebut akan diambil 4 nilai keluaran yang akan digunakan sebagai masukan pada proses selanjutnya yaitu 4 nilai tertinggi dari hasil proses *autocorrelation.* Berikut alur proses dari ekstraksi fitur ritme seperti yang ditunjukkan pada gambar 3.10.



Gambar 3. 10 Alur Proses Ekstraksi Fitur Ritme

1. *Discrete Wavelet Transform*

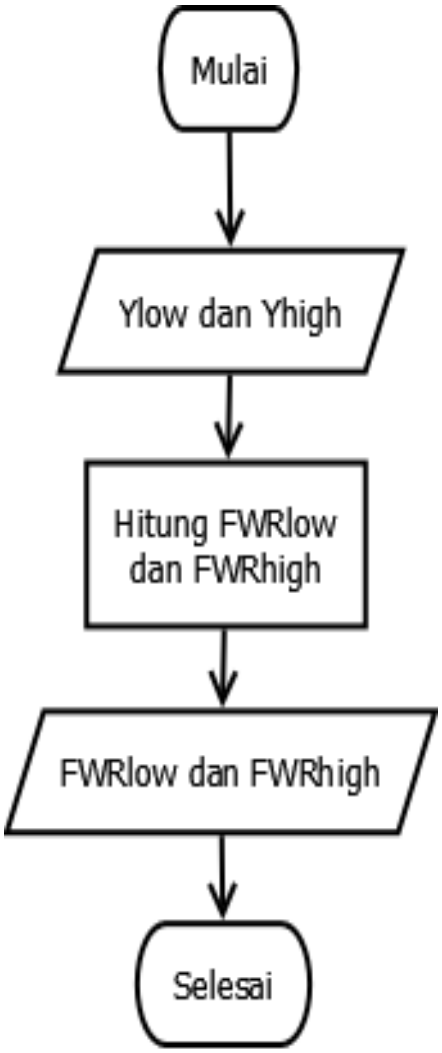
Tahapan pertama dari fitur ritme adalah *Discrete Wavelet Transform. Discrete Wavelet Transform* (DWT) merupakan salah satu bentuk diskrit dari wavelet dan seringkali digunakan. DWT akan mengubah sinyal sumber menjadi dua klasifikasi sinyal yaitu frekuensi tinggi serta frekuensi rendah. Berikut alur proses dari DWT seperti yang ditunjukkan pada gambar 3.11.



Gambar 3. 11 Alur Proses DWT

2. *Full Wave Rectification*

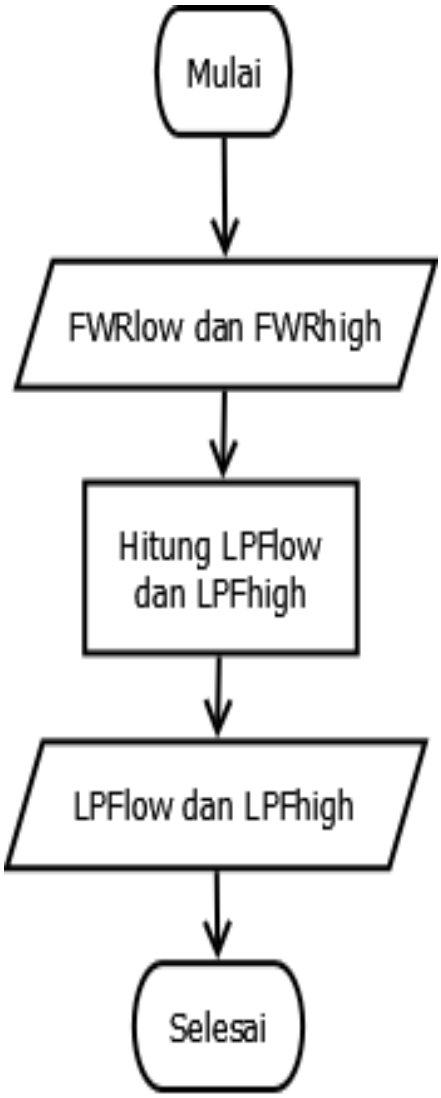
Tahapan kedua pada fitur ritme adalah *Full Wave Rectification.* Setelah mendapatkan nilai dari proses dwt yang berupa y_{low} dan y_{high} , proses selanjutnya yang akan dilakukan adalah perhitungan FWR. Berikut alur proses dari FWR seperti yang ditunjukkan pada gambar 3.12.



Gambar 3. 12 Alur Proses FWR

3. *Low Pass Filtering*

Tahapan ketiga pada fitur ritme adalah *Low Pass Filtering.* Setelah mendapatkan nilai dari proses FWR yang berupa FWR_{low} dan FWR_{high}, proses selanjutnya yang akan dilakukan adalah perhitungan LPF. Berikut alur proses dari LPF seperti yang ditunjukkan pada gambar 3.13.



Gambar 3. 13 Alur Proses LPF

4. *Downsampling*

Tahapan keempat pada fitur ritme adalah *Downsampling.* Setelah mendapatkan nilai dari proses LPF yang berupa LPF_{low} dan LPF_{high}, proses selanjutnya yang akan dilakukan adalah perhitungan *Downsampling*. Berikut alur proses dari *Downsampling* seperti yang ditunjukkan pada gambar 3.14.

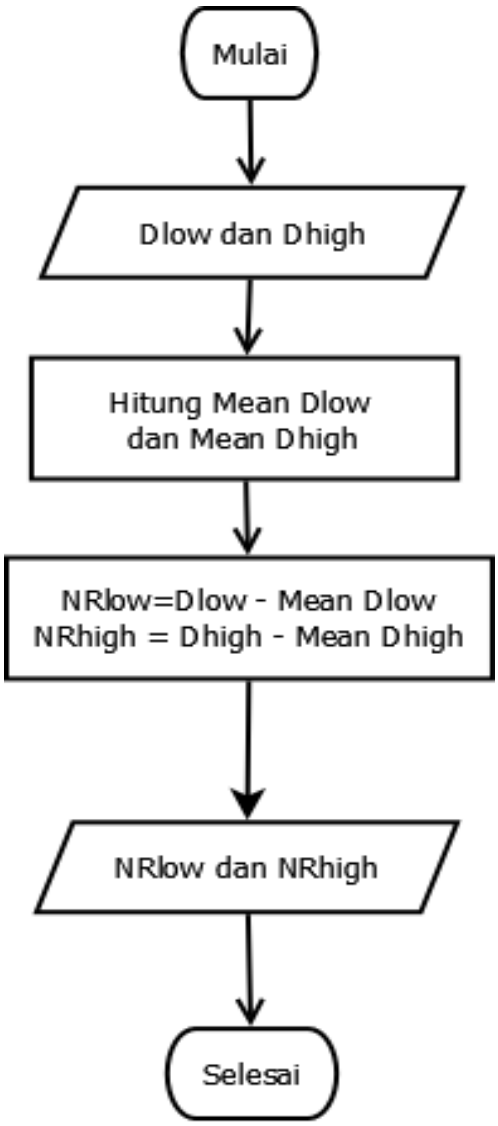


Gambar 3. 14 Alur Proses *Downsampling*

5. *Mean Removal*

Tahapan kelima pada fitur ritme adalah *Mean Removal* atau *Normalizaton* (NR). Setelah mendapatkan nilai dari proses *Downsampling* yang berupa D_{low} dan D_{high}, proses selanjutnya yang akan dilakukan adalah perhitungan *Mean*

Removal. Berikut alur proses dari *Mean Removal* seperti yang ditunjukkan pada gambar 3.15.

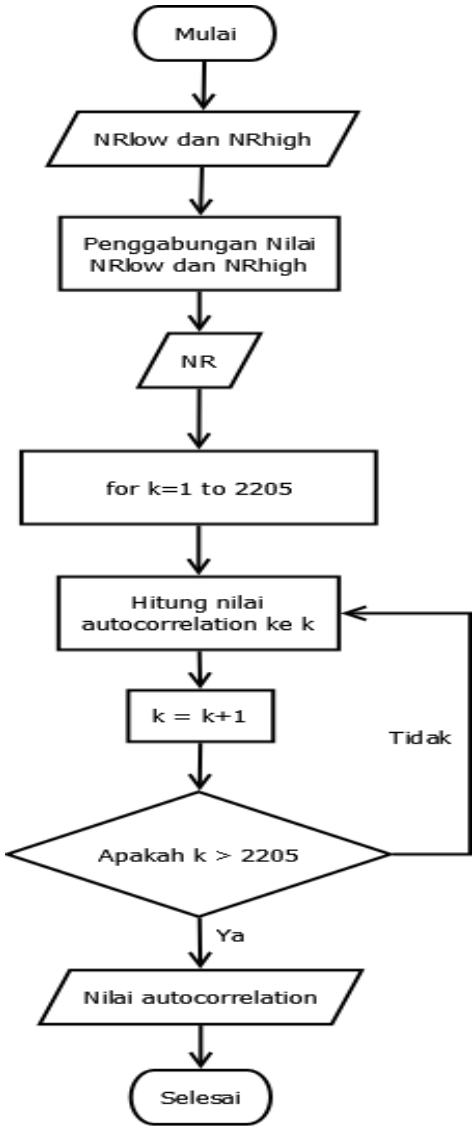


Gambar 3. 15 Alur Proses *Mean Removal*

6. *Autocorrelation*

Tahapan keenam pada fitur ritme adalah *Autocorrelation*(AR). Setelah mendapatkan nilai dari proses *Mean Removal* yang berupa NR_{low} dan NR_{high}, proses selanjutnya yang akan dilakukan adalah melakukan penggabungan nilai

Mean Removal. Setelah digabungkan, maka selanjutnya akan dihitung nilai dari *Autocorrelation*. Berikut alur proses dari *Autocorrelation* seperti yang ditunjukkan pada gambar 3.16.



Gambar 3. 16 Alur Proses *Autocorrelation*

7. Pemilihan 4 titik tertinggi

Tahapan ketujuh pada fitur ritme adalah pemilihan 4 titik tertinggi. Setelah mendapatkan nilai dari proses *Autocorrelation*, proses selanjutnya yang akan dilakukan adalah melakukan pengurutan nilai *Autocorrelation* dari nilai yang

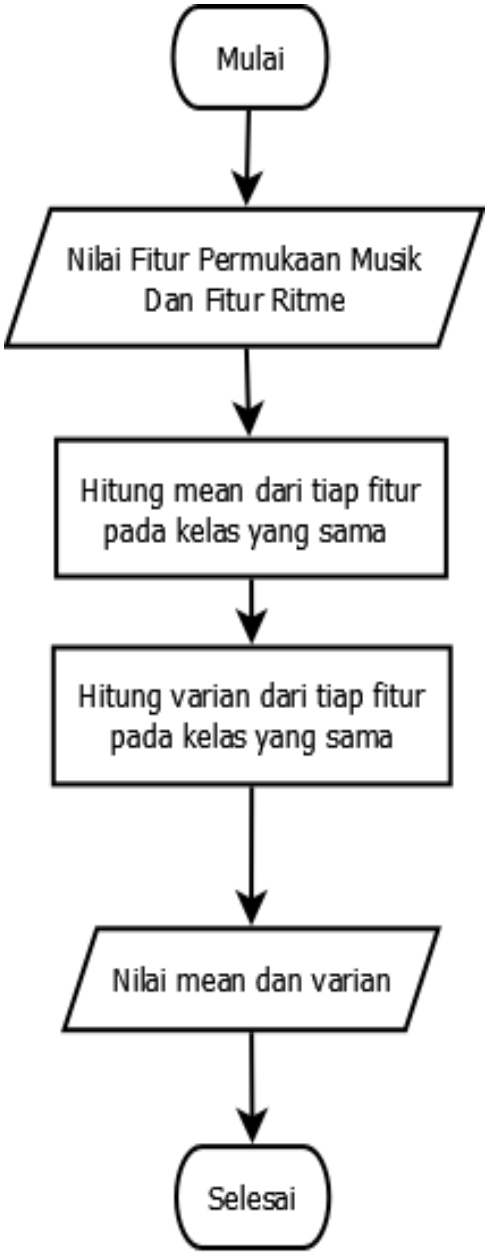
paling tinggi ke yang rendah. Lalu akan diambil 4 nilai tertinggi yaitu Peak1, Peak2, Peak3 dan Peak4. Nilai tersebut merupakan nilai keluaran dari ekstraksi fitur ritme. Alur proses pemilihan 4 titik tertinggi ditunjukkan pada gambar 3.17.



Gambar 3. 17 Alur Proses Pemilihan 4 Titik Tertinggi

3.2.2 Analisis Pelatihan

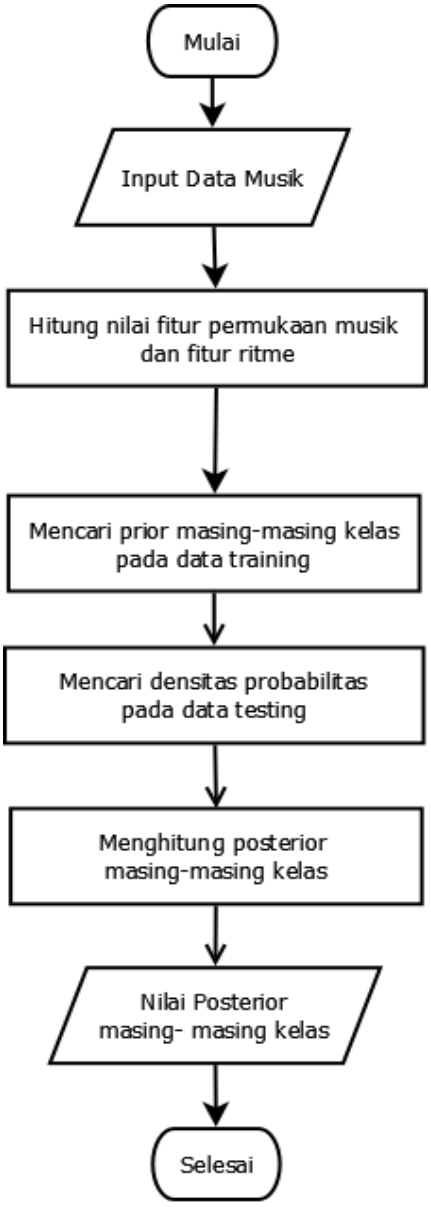
Pelatihan dengan naive bayes dilakukan untuk mendapatkan data latih, proses ini dilakukan dengan melakukan perhitungan nilai mean dan varians. Berikut adalah alur proses pelatihan dengan naive bayes seperti yang ditunjukkan pada gambar 3.18.



Gambar 3. 18 Alur Pelatihan

3.2.3 Analisis Pengujian

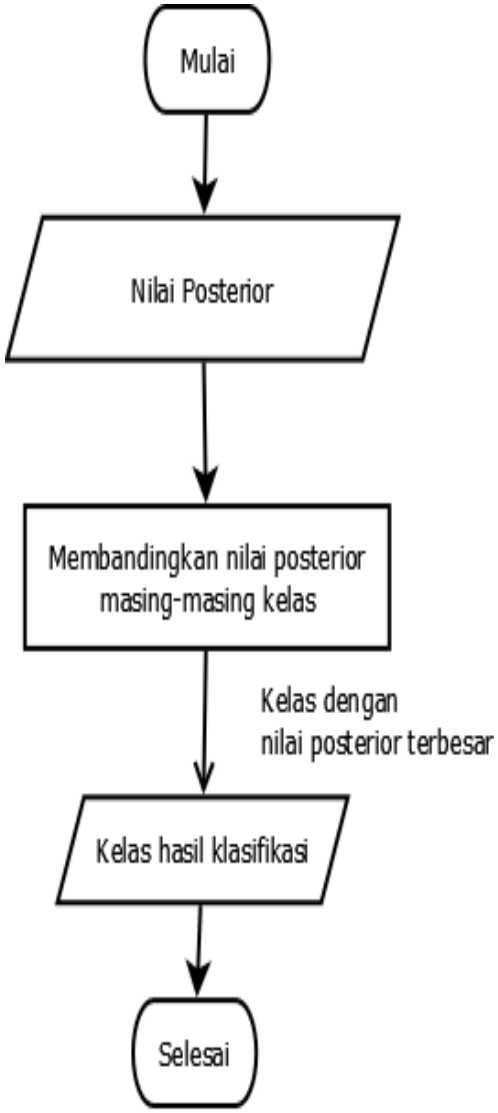
Pengujian adalah tahapan pada proses klasifikasi genre musik untuk menghitung berdasarkan klasifikasi naive bayes. Pada proses ini data musik yang dimasukan dapat dihasilkan nilai probabilitasnya. Berikut adalah alur proses pengujian dengan naive bayes seperti yang ditunjukkan pada gambar 3.19.



Gambar 3. 19 Alur pengujian

3.2.4 Analisis Data Keluaran

Analisis data keluaran adalah tahap akhir yang akan dilakukan. yaitu mencari nilai terbesar terhadap nilai probabilitas yang didapatkan pada saat analisis pengujian. Berikut adalah alur proses pengujian dengan naive bayes seperti yang ditunjukkan pada gambar 3.20.



Gambar 3. 20 Alur Analisis Data Keluaran

3.3 Analisis Perhitungan

Analisis perhitungan klasifikasi genre musik yang dibangun memiliki 4 tahapan yaitu analisis data masukan, analisis pelatihan, analisis pengujian dan analisis data keluaran.

3.3.1 Analisis Data Masukan

Data musik yang digunakan berdurasi 5 detik dan sampling ratenya adalah 22050 Hz. Sampling rate 22050 Hz berarti terdapat 22050 titik sinyal dalam satu detik.

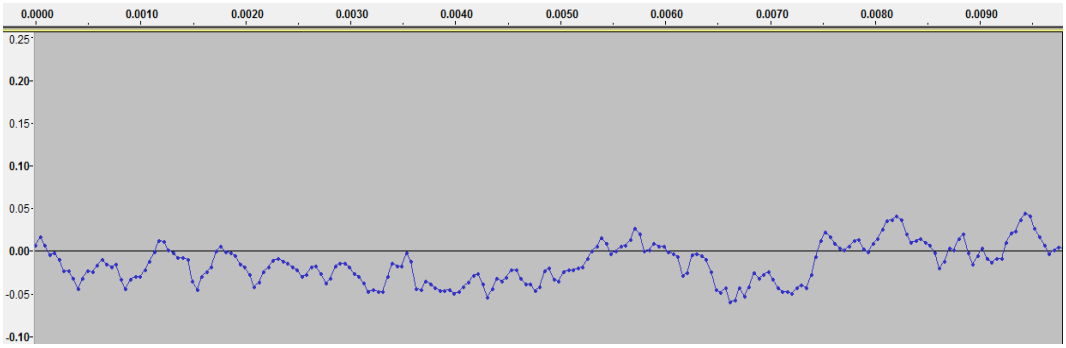
Jumlah titik sinyal (5 detik) = 22050 * 5 = 110250 titik sinyal.

3.3.1.1 *Analysis Window*

Pada tahap *analysis window* nantinya data musik yang berdurasi 5 detik dan jumlah titik sinyalnya 110250 titik sinyal, akan dipotong menjadi 250 *analysis window* sehingga :

Jumlah titik sinyal (0.02 detik) = 110250 / 250 = 441 titik sinyal.

Ilustrasi dari titik sinyal pada *analysis window* ditunjukkan pada gambar 3.21.



Gambar 3. 21 Titik Sinyal

Nilai dari 441 titik sinyal pada 250 *analysis window* tersebut dapat dilihat pada tabel 3.1. Untuk melihat nilai lengkap dari *analysis window* 1 dengan 441 titik sinyal dapat dilihat pada lampiran f.

Tabel 3. 1 Nilai Sinyal Pada Tiap Analysis Window

Window f(n)	*Analysis Window* 1	*Analysis Window 2*	*Analysis Window 3*	…	*Analysis Window* 250
0	0.0073	-0.0086	0.0456	…	-0.0044
1	0.0166	-0.0020	0.0483	…	0.0053
2	0.0076	0.0063	0.0419	…	0.0053
…	…	…	…	…	…

440	-0.0131	0.0342	-0.0600	…	-0.1056

3.3.1.2 *Fast Fourier Transform* (FFT)

Fast Fourier Transform adalah suatu algoritma yang digunakan untuk merepresentasikan sinyal dalam domain waktu diskrit dan domain frekuensi.

Dengan menggunakan persamaan (2.2) maka:

$F_{[k]} = \sum_{n=0}^{440} f(n)\cos(2\pi kn/N) - j\sum_{n=0}^{440} f(n)\sin(2\pi kn/N)$

Dimana:

f(n) = sinyal suara ke-n

N= jumlah sampel

$F_{[k]}$ = hasil dari FFT ke-k

Setelah ditemukan nilai FFTnya maka dicari nilai *magnitude*-nya:

$M_{[k]} = [R^2 + I^2]^{1/2}$

Dimana:

$M_{[k]}$= nilai *magnitude* ke-k

Untuk *analysis window* 1 dengan data nilai sinyal (0.0073, 0.0166, 0.0076, … , -0.0131) seperti yang ditunjukkan pada tabel 3.1, perhitungan nilai FFTnya adalah:

$F_{[0]}$ = [0.0073 $(\cos(\frac{2\pi*0*0}{441}))$] – j (0.0073) sin $(\frac{2\pi*0*0}{441})$ +

[0.0166 $(\cos(\frac{2\pi*0*1}{441}))$] – j (0.0166) sin $(\frac{2\pi*0*1}{441})$ +

[0.0076 $(\cos(\frac{2\pi*0*2}{441}))$] – j (0.0076) sin $(\frac{2\pi*0*2}{441})$ + … +

[-0.0131 $(\cos(\frac{2\pi*0*440}{441}))$] – j (-0.0131) sin $(\frac{2\pi*0*440}{441})$

= -0.9577 + 0j

$M_{[0]} = [R^2 + I^2]^{1/2} = [(-0.9577)^2 + 0^2]^{1/2} = 0.9577$

Lakukan perhitungan hingga $F_{[440]}$ dengan cara yang sama, maka akan menghasilkan nilai: (0.9577, 4.7056, 1.4848, … , 4.7056)

Dari perhitungan diatas diperoleh 441 nilai *magnitude* dari hasil FFT pada 250 *analysis window* seperti yang ditunjukkan pada tabel 3.2.

Tabel 3. 2 Nilai Magnitude Pada Tiap Analysis Window

Window F(n)	*Analysis Window* 1	*Analysis Window 2*	*Analysis Window 3*	…	*Analysis Window* 250
0	0.9577	1.9039	0.0237	…	1.6922
1	4.7056	3.8953	6.1830	…	6.3740
2	1.4848	2.2395	6.7796	…	3.8601
…	…	…	…	…	…
440	4.7056	3.8953	6.1830	…	6.3740

Hasil nilai dari magnitude tersebut akan dipergunakan pada tahapan berikutnya.

3.3.1.3 Ekstraksi Fitur Permukaan Musik

Ekstraksi fitur permukaan musik merupakan proses untuk mendapatkan ciri atau fitur yang terdapat pada musik. Data musik yang telah dihitung dengan fft dan didapatkan nilai *magnitude-nya* akan digunakan pada tahap ini, Tahapan ini akan menghitung nilai *centroid, rolloff, flux, zerocrossings* dan *low energy.* Dari perhitungan tersebut akan diambil 9 nilai keluaran yang akan digunakan sebagai masukan pada proses selanjutnya yaitu *mean-centroid, std-centroid, mean-rolloff, std-rolloff, mean-flux, std-flux, mean-zerocrossings, std-zerocrossings,* dan *low energy*.

1. *Spectral Centroid*

Setelah didapatkan nilai magnitude dari setiap analysis window, maka selanjutnya akan dihitung nilai centroid. Nilai keluaran yang akan digunakan pada proses selanjutnya adalah nilai mean-centroid dan std-centroid.

Untuk menghitung nilai centroid maka digunakan persamaan (2.4) yaitu:

$$C = \sum_{k=0}^{440} kM_{[k]} / \sum_{k=0}^{440} M_{[k]}$$

Dimana:

C= Nilai *centroid*

k = 0,…,440 adalah *frequency bin*

$M_{[k]}$ = Nilai magnitude yang diperoleh dari FFT

Dari nilai *magnitude* pada *analysis window* 1 yang terdapat pada tabel 3.2, diperoleh perhitungan sebagai berikut:

Tabel 3. 3 Perhitungan *Centroid* Pada Analysis Window Ke 1

k	$M_{[k]}$	k*$M_{[k]}$
0	0.9577	1.9039
1	4.7056	3.8953
2	1.4848	2.2395
…	…	…
440	4.7056	3.8953
Jumlah	115.8696	25338.0745
C = 25338.0745/115.8696 = 218.6774		

Dari tabel 3.3 diperoleh nilai *Centroid* = 218.6774 . Nilai *centroid* tersebut merupakan nilai *centroid* untuk *analysis window* 1. Lakukan hal yang sama pada 249 *analysis window* yang lain. Hasil nilai *centroid* pada 250 *analysis windows* ditunjukkan pada tabel 3.4 dan untuk nilai *centroid* lengkap pada 250 *analysis window*, dapat dilihat pada lampiran f.

Tabel 3. 4 Nilai *Centroid* pada 250 *Analysis Windows*

Analysis Window	Nilai *Centroid*
1	218.6774
2	216.7712
3	220.4706
…	…
250	219.3471

Untuk menghitung nilai *mean-centroid* maka semua nilai dari *centroid* harus dijumlahkan dan dibagi dengan jumlah *analysis window.* Maka hasil nilai *mean-centroid* adalah sebagai berikut:

μ = (218.6774+216.7712+220.4706+…+219.347)/250

= 218.6529

Untuk menghitung standar deviasi *(*std) pada *centroid*, rumus standar deviasi adalah seperti pada persamaan (3.1):

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}} \quad (3.1)$$

Dimana:

$\sigma =$ Standar deviasi

μ = nilai rata-rata

$y_i =$ nilai *centroid* ke-i

Proses perhitungan nilai std-*centroid* ditunjukkan pada tabel 3.5 sebagai berikut:

Tabel 3. 5 Perhitungan Std-*Centroid*

i	y_i	μ	$y_i - \mu$	$(y_i - \mu)^2$
1	218.6774	218.6529	0.0245	0.0006
2	216.7712	218.6529	-1.8817	3.5407
3	220.4706	218.6529	1.8177	3.3041
…	…	…	…	…
250	219.347	218.6529	0.6942	0.4819
Jumlah				526.6677

$$\sigma = \sqrt{\frac{\sum_{i=1}^{250}(y_i-\mu)^2}{n-1}} = \sqrt{\frac{526.6677}{249}} = 1.4543$$

Dari perhitungan diatas didapat nilainya adalah:

Std-centroid = 1.4543

2. *Spectral Rolloff*

Spectral rolloff merupakan frekuensi dibawah 85% dari energi pada *magnitude spectrum* yang terkonsentrasi. *Spectral rolloff* merupakan keseimbangan distribusi energy diantara frekuensi tinggi dan rendah pada spektrum. *Spectral rolloff* dihitung dengan persamaan (2.5) yaitu:

$$\sum_0^R M_{[k]} = 0.85 \sum_0^{440} M_{[k]}$$

Dimana:

$M_{[k]}$ = nilai magnitude

$\sum_0^R M_{[k]}$ = jumlah nilai magnitude dikali 0.85

Nilai magnitude dari 441 sampel pada *analysis window 1* adalah seperti yang terdapat pada tabel 3.2:

(0.9577, 4.7056, 1.4848, … , 4.7056)

Dengan persamaan (2.5) perhitungannya sebagai berikut:

$\sum_0^R M_{[k]} = 0.85 \sum_0^{440} M_{[k]}$

$\sum_0^R M_{[k]}$ = 0.85 (0.9577+4.7056+1.4848+ … +4.7056)

$\sum_0^R M_{[k]}$ = 0.85 (115.8696)

$\sum_0^R M_{[k]}$ = 98.4892

Dari perhitungan diatas diketahui bahwa nilai $\sum_0^R M[k]$ = 98.4892.

Untuk mencari nilai R akan melalui beberapa tahap perulangan untuk mengetahui pada data keberapa nilai total sebelah kiri lebih besar dari nilai sebelah kanan.

Untuk mencari posisi nilai R, perhitungannya ditampilkan pada tabel 3.6.

Tabel 3. 6 Perhitungan Untuk Mencari Posisi Nilai R

k	$M_{[k-1]}$	energy	$\sum_0^R M_{[k]}$	energy <= $\sum_0^R M_{[k]}$	energy = energy + $M_{[k-1]}$
1	0.9577	0	98.4892	True	0.9577
2	4.7056	0.9577	98.4892	True	5.6633
3	1.4848	5.6633	98.4892	True	7.1482
…	…	…	…	…	…
434	2.7797	98.2871	98.4892	True	101.0669
435	1.9352	101.0669	98.4892	False	103.0022

Karena pada tahap 435 nilai energy lebih besar dari 98.4892, maka selanjutnya akan dihitung nilai *Rolloff*.

Rolloff = 435/441 = 0.9864

Dimana nilai 435 menggambarkan tahapannya dan nilai 441 adalah total dari seluruh sinyal.

Nilai *rolloff* = 0.9864 merupakan nilai *rolloff* untuk 1 *analysis window*. Lakukan hal yang sama pada 249 *analysis window* yang lain. Hasil nilai *rolloff* pada 250 *analysis window* ditunjukkan pada tabel 3.7 dan nilai lengkapnya pada lampiran f .

Tabel 3. 7 Nilai *Rolloff* Pada 250 A*nalysis Window*

Analysis Window	Nilai *Rolloff*
1	0.9864
2	0.9841
3	0.9909
…	…
250	0.9864

Untuk menghitung nilai *mean-rolloff* maka semua nilai dari *rolloff* harus dijumlahkan dan dibagi dengan jumlah *analysis window*. Nilai *mean-rolloff* adalah sebagai berikut:

Mean-rolloff = (0.9864 + 0.9841 + 0.9909 + … + 0.9864) / 250

= (245.8934) / 250 = 0.9836

Untuk menghitung standar deviasi *(*std) pada *rolloff*, rumus standar deviasi seperti pada persamaan (3.1) adalah:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}}$$

Dimana:

$\sigma =$ Standar deviasi

μ = nilai rata-rata

$y_i =$ nilai rolloff ke-i

Proses perhitungan nilai std-*rolloff* ditunjukkan pada tabel 3.8 sebagai berikut:

Tabel 3. 8 Perhitungan Std-*Rolloff*

Rolloff	y_i	μ	$y_i - \mu$	$(y_i - \mu)^2$
1	0.9864	0.9836	0.0028	0.0000080
2	0.9841	0.9836	0.0006	0.0000003
3	0.9909	0.9836	0.0074	0.0000541
…	…	…		…
250	0.9864	0.9836	0.0028	0.0000080
Jumlah				0.0175653

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}} = \sqrt{\frac{0.0175653}{249}} = 0.0084$$

Dari perhitungan diatas didapat:

Std-rolloff = 0.0084

3. *Spectral Flux*

Flux adalah ukuran dari *spectral change. Flux* menunjukkan berapa banyak perubahan spektral dari waktu ke waktu. Persamaan yang digunakan sesuai dengan persamaan (2.6) adalah:

F= $\| M_{[k]} - Mp_{[k]} \|$

Berdasarkan [7], k= indeks dari *analysis window*. $M_{[k]}$ merupakan nilai magnitude pada $M_{[k]}$ dan $Mp_{[k]}$ nilai magnitude saat k-1. Dalam *flux*, dicari perubahan energy dari satu *window* dengan *window* sebelumnya. Namun pada kasus k=0, energy pada *window* tersebut diurangi dengan nilai 0. Sehingga hasilnya adalah *window* itu sendiri. Selanjutnya pada k=1,…,N-1, *energy window* ke- k akan dikurangi *energy* pada *window* sebelumnya. Kemudian tiap hasil pengurangan akan dikuadradkan. Lalu hasil kuadrat dari tiap titik tersebut akan dijumlahkan. Proses perhitungan *flux* akan ditunjukkan pada tabel 3.9.

Tabel 3. 9 Perhitungan *Flux* Pada *Analysis Window* 1

$M_{[k]}$	$Mp_{[k]}$	$M_{[k]} - Mp_{[k]}$	$(M_{[k]} - Mp_{[k]})^2$
0.9577	0	0.9577	0.9172
4.7056	0.9577	3.7479	14.0464
1.4848	4.7056	-3.2207	10.3732
…	…	…	…
4.7056	1.4848	3.2207	10.3732
Jumlah			84.7909

Dari perhitungan diatas, dapat dilihat nilai Spectral Flux dari *Analysis window 1* yaitu:

Flux = 84.7909

Dengan contoh perhitungan seperti diatas, akan dilakukan perhitungan nilai flux untuk 249 *analysis window* yang lain seperti yang ditunjukkan pada tabel 3.10 dan nilai lengkapnya pada lampiran f.

Tabel 3. 10 Nilai *Flux* Untuk 250 *Analysis Window*

Analysis Window	Nilai Flux
1	84.7909
2	75.9012
3	310.7445
…	…
250	966.8865

Tahapan selanjutnya adalah mencari nilai *mean-flux* dan *std-flux.*

Nilai *mean-flux* adalah sebagai berikut:

Mean-flux = (84.7909 + 75.9012 + 310.7445 + … + 966.8865) / 250

= (655470.4260) / 250 = 2621.8817

Rumus standar deviasi adalah:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}}$$

Dimana:

$\sigma =$ Standar deviasi

μ = nilai rata-rata

$y_i =$ nilai *flux* ke-i

Proses perhitungan nilai std-*flux* ditunjukkan pada tabel 3.11 sebagai berikut:

Tabel 3. 11 Perhitungan Std-*Flux*

i	y_i	μ	$y_i - \mu$	$(y_i - \mu)^2$
1	84.7909	2621.8817	-2537.1	6436829.791
2	75.9012	2621.8817	-2546	6482016.823
3	310.7445	2621.8817	-2311.1	5341355.049
…	…	…	…	…
250	966.8865	2621.8817	-1655	2739009.168
Jumlah				3162534784.4620

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}} = \sqrt{\frac{3162534784.4620}{249}} = 3563.8382$$

Dari perhitungan diatas didapat:

Std-flux = 3563.8382

4. *Zerocrossings*

Zerocrossings adalah jumlah dari penyebrangan sinyal dalam 1 *window* yang melalui nilai 0.

Untuk proses perhitungannya akan dicoba pada *analysis window* 1:

Tabel 3. 12 Perhitungan *Zerocrossings* pada *Analysis Window* 1

f(n)	Nilai Sinyal	Nilai *Zerocrossing*
0	0.0073	0
1	0.0166	0
2	0.0076	0
3	- 0.0035	1
…	…	
437	0.0056	50
438	- 0.0220	51
439	- 0.0239	51
440	- 0.0131	51
Nilai Zerocrossing		51

Dari perhitungan pada tabel 3.12 diatas didapat nilai *zerocrossing* adalah 51 yang berarti ada 51 kali sinyal melewati titik 0.

Selanjutnya dilakukan perhitungan untuk 249 *analysis window* lainnya dan didapat nilainya seperti yang ditunjukkan pada tabel 3.13 dan untuk melihat nilai lengkapnya pada lampiran f.

Tabel 3. 13 Nilai *Zerocrossings*

Analysis Window	Nilai *Zerocrossing*
1	51
2	67
3	35
…	…
250	22

Nilai *mean-zerocrossings* adalah sebagai berikut:

Mean-zero = (51 + 67 + 35 + … + 22) / 250

= 9299 / 250 = 37.1960

Untuk menghitung standar deviasi *(*std) pada *zerocrossings*, rumus standar deviasi adalah:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}}$$

Dimana:

$\sigma =$ Standar deviasi

μ = nilai rata-rata

$y_i =$ nilai *zerocrossing* ke-i

Proses perhitungan nilai std-*zerocrossings* ditunjukkan pada tabel 3.14 sebagai berikut:

Tabel 3. 14 Perhitungan *Std-Zerocrossing*

i	y	μ	$y-\mu$	$(y-\mu)^2$
1	51	37.196	13.804	190.5504
2	67	37.196	29.804	888.2784
3	35	37.196	-2.196	4.8224
…	…	…	…	…
250	22	37.196	-15.196	230.9184
Jumlah				65591.3960

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\mu)^2}{n-1}} = \sqrt{\frac{65591.3960}{249}} = 16.2302$$

Dari perhitungan diatas didapat:

Std-zero = 16.2302

5. *Low Energy*

LowEnergy adalah persentase dari jumlah *analysis windows* yang memiliki energi kurang dari rataan seluruh *analysis windows* dalam satu data musik. Perhitungan nilai rata-rata *magnitude* ditunjukkan pada tabel 3.15.

Tabel 3. 15 Perhitungan Nilai Rata-Rata *Magnitude*

Window F(n)	*Analysis Window* 1	*Analysis Window 2*	*Analysis Window 3*	…	*Analysis Window* 250
0	0.9577	1.9039	0.0237	…	1.6922
1	4.7056	3.8953	6.1830	…	6.3740
2	1.4848	2.2395	6.7796	…	3.8601
…	…	…	…	…	…
440	4.7056	3.8953	6.1830	…	6.3740
Energy	115.8696	112.5889	178.0661		323.6532
Rata-rata *Energy*	432.0300				

Untuk mencari nilai dari *lowenergy*, nilai *energy* akan dibandingkan dengan rata-rata. Lalu akan dihitung jumlah *energy* yang lebih kecil dari rata-rata tersebut. Nilai *lowenergy* didapat dari jumlah hasil hitungnya dibagi 250. Perhitungan nilai *Low Energy* ditunjukkan pada tabel 3.16

Tabel 3. 16 Perhitungan Nilai *LowEnergy*

i	*Energy*	Rata-rata *Energy*	*Energy* < rata-rata	Jumlah
1	115.8696	432.0300	True	1
2	112.5889	432.0300	True	2
…	…			
244	213.6112	432.0300	True	144
245	504.0121	432.0300	False	144
246	546.4475	432.0300	False	144
247	415.9578	432.0300	True	145
248	361.6244	432.0300	True	146
249	360.5225	432.0300	True	147
250	323.6532	432.0300	True	148
LowEnergy				0.5920

Dari perhitungan diatas didapat:

LowEnergy = 0.5920

3.3.1.4 Ekstraksi Fitur Ritme

Fitur ritme dikarakterisasikan sebagai pergerakan sinyal musik terhadap waktu dan mengandung informasi seperti yang biasanya terdapat pada ritme, *beat, tempo* dan *time signature*. Fitur yang diset untuk merepresentasikan struktur ritme berasal dari mendeteksi periode yang paling penting pada sebuah sinyal. Ekstraksi fitur ritme merupakan proses untuk mendapatkan ciri atau fitur yang terdapat pada musik., Tahapan ini akan menghitung nilai *dwt, full wave rectification, low pass filtering, downsampling, mean removal* dan *autocorrelation.* Dari perhitungan tersebut akan diambil 4 nilai keluaran yang akan digunakan sebagai masukan pada proses selanjutnya yaitu 4 nilai tertinggi dari hasil proses *autocorrelation.*

1. *Discrete Wavelet Transform* (DWT)

Discrete Wavelet Transform (DWT) merupakan salah satu bentuk diskrit dari wavelet dan seringkali digunakan. DWT akan mengubah sinyal sumber menjadi dua klasifikasi sinyal yaitu frekuensi tinggi serta frekuensi rendah. Proses DWT diawali dengan proses filtering dengan melewatkan sinyal pada seperangkat highpass filter dan lowpass filter.Pada penelitian kali ini akan menggunakan DAUB4 sebagai proses filternya

Filter yang akan digunakan adalah Daubechies 4 seperti yang ditunjukkan pada persamaan (3.2) dan (3.3) yang bernilai:

$$\begin{aligned} h(0) &= 1 + \sqrt{3} / 4\sqrt{2} \\ h(1) &= 3 + \sqrt{3} / 4\sqrt{2} \\ h(2) &= 3 - \sqrt{3} / 4\sqrt{2} \\ h(3) &= 1 - \sqrt{3} / 4\sqrt{2} \end{aligned} \tag{3.2}$$

$$\begin{aligned} g(0) &= 1 - \sqrt{3} / 4\sqrt{2} \\ g(1) &= -3 + \sqrt{3} / 4\sqrt{2} \\ g(2) &= 3 + \sqrt{3} / 4\sqrt{2} \\ g(3) &= -1 - \sqrt{3} / 4\sqrt{2} \end{aligned} \tag{3.3}$$

h = [h0,h1,h2,h3]

g = [g0,g1,g2,g3]

Nilai h dan g merupakan nilai dari *scaling function* dan *wavelet function* yang dijabarkan pada [20]. Nilai dari h dan g akan dihitung nilai y_{low} dan y_{high}. Jika data musiknya memiliki N nilai maka pada perjitungan dwtnya akan dihitung pada N/2 nilai.

Untuk n = 0 sampai 55124 lakukan perhitungan nilai $y_{low}(n)$ sebagai berikut:

$$y_{low}(n) = \text{h0} * \text{f(n)} + \text{h1} * \text{f(n+1)} + \text{h2} * \text{f(n+2)} + \text{h3} * \text{f(n+3)}$$

$y_{low}(0)$ = ((1+$\sqrt{3}$/4$\sqrt{2}$) * 0.0073) + ((3+$\sqrt{3}$/4$\sqrt{2}$) * 0.0166) +
((3-$\sqrt{3}$/4$\sqrt{2}$) * 0.0076) + ((1-$\sqrt{3}$/4$\sqrt{2}$) * -0.0035)

= (0.4830*0.0073)+(0.8365*0.0166)+(0.2241*0.0076)+(-0.1294*-0.0035)

= 0.0196

Dengan cara yang sama lakukan untuk nilai n=1 sampai 55124 maka akan didapatkan nilai y_{low} seperti yang ditunjukkan tabel 3.17:

Tabel 3. 17 Nilai Y(*low*)

n	$y_{low}(n)$
0	0.0196
1	0.0015
2	-0.0111
…	…
55124	-0.1412

Untuk n = 0 sampai 55124 lakukan perhitungan nilai $y_{high}(n)$ sebagai berikut:

$$y_{high}(n) = \text{g0} * \text{f(n)} + \text{g1} * \text{f(n+1)} + \text{g2} * \text{f(n+2)} + \text{g3} * \text{f(n3)}$$

$y_{high}(0)$ = ((1-$\sqrt{3}$/4$\sqrt{2}$) * 0.0073) + ((-3+$\sqrt{3}$/4$\sqrt{2}$) * 0.0166) +
((3+$\sqrt{3}$/4$\sqrt{2}$) * 0.0076) + ((-1-$\sqrt{3}$/4$\sqrt{2}$) * -0.0035)

= (-0.1294*0.0073)+(-0.2241*0.0166)+(0.8365*0.0076)+(-0.4830*-0.0035)

= 0.0034

Dengan cara yang sama lakukan untuk nilai n=1 sampai 55124 maka akan didapatkan nilai y_{high} seperti yang ditunjukkan pada tabel 3.18 yaitu:

Tabel 3. 18 Nilai Y(*high*)

n	$y_{high}\,(n)$
0	0.0034
1	0.0024
2	-0.0058
…	…
55124	0.0358

2. *Full Wave Rectification* (FWR)

Setelah mendapatkan nilai dari proses dwt yang berupa y_{low} dan y_{high}, proses selanjutnya yang akan dilakukan adalah perhitungan FWR. *Full Wave Rectification* dihitung dengan persamaan (2.7):

FWR[n] = abs (y[n])

Untuk nilai dari $y_{low}\,(n)$ maka nilai FWRlow-nya ditunjukkan pada tabel 3.19.

Tabel 3. 19 Nilai FWRlow

n	$y_{low}\,(n)$	FWRlow(n)
0	0.0196	0.0196
1	0.0015	0.0015
2	-0.0111	0.0111
…	…	…
55124	-0.1412	0.1412

Untuk nilai dari $y_{high}(n)$ maka nilai FWRhigh-nya ditunjukkan pada tabel 3.20.

Tabel 3. 20 Nilai FWRhigh

n	$y_{high}\,(n)$	FWRhigh(n)
0	0.0034	0.0034
1	0.0024	0.0024
2	-0.0058	0.0058
…	…	…
55124	0.0358	0.0358

3. *Low Pass Filtering* (*One Pole Filter with* = 0.99)

Setelah mendapatkan nilai dari proses FWR yang berupa $FWRlow$ dan $FWRhigh$, proses selanjutnya yang akan dilakukan adalah perhitungan LPF. Low pass filtering (LPF) dihitung menggunakan persamaan (2.8):

LPF[n] = (1-α) FWR[n] – αFWR[n]

Dimana:

α = 0.99

Untuk nilai dari $FWRlow(n)$ maka nilai LPFlow-nya ditunjukkan pada tabel 3.21.

Tabel 3. 21 Nilai LPFlow

n	FWRlow(n)	$LPFlow$(n)
0	0.0196	-0.0192
1	0.0015	-0.0014
2	0.0111	-0.0109
…	…	…
55124	0.1412	-0.1383

Untuk nilai dari $FWRhigh(n)$ maka nilai LPFhigh-nya ditunjukkan pada tabel 3.22.

Tabel 3. 22 Nilai LPFhigh

n	FWRhigh(n)	$LPFhigh$(n)
0	0.0034	-0.0033
1	0.0024	-0.0024
2	0.0058	-0.0057
…	…	…
55124	0.0358	-0.0351

4. *Downsampling*

Setelah mendapatkan nilai dari proses LPF yang berupa $LPFlow$ dan $LPFhigh$, proses selanjutnya yang akan dilakukan adalah perhitungan *downsampling*. Downsampling (↓) dihitung dengan persamaan (2.9):

D[n] = LPF[kn]

Dimana:

k=4

Untuk nilai dari $LPFlow(n)$ maka nilai Dlow-nya ditunjukkan pada tabel 3.23, dimana Dlow[n] = LPFlow[4n]

Tabel 3. 23 Nilai Dlow

n	$Dlow$(n)
0	-0.0192
1	-0.0544
2	-0.0231
…	…
13782	-0.1383

Untuk nilai dari $LPFhigh(n)$ maka nilai Dhigh-nya ditunjukkan pada tabel 3.24, dimana Dhigh[n] = LPFhigh[4n]

Tabel 3. 24 Nilai Dhigh

n	$Dhigh$(n)
0	-0.0033
1	-0.0017
2	-0.0001
…	…
13782	-0.351

5. *Mean Removal*

Setelah mendapatkan nilai dari proses *downsampling* yang berupa $Dlow$ dan $Dhigh$, proses selanjutnya yang akan dilakukan adalah perhitungan *mean removal*.*Normalization* (NR) atau *mean removal* dihitung menggunakan persamaan (2.10):

NR[n] = D[n] – E D[n]

Dimana:

E D[n] = nilai rata-rata dari D[n]

Untuk nilai dari $Dlow(n)$ maka nilai NRlow-nya ditunjukkan pada tabel 3.25, dimana NRlow[n] = Dlow[n] – E Dlow[n]

Tabel 3. 25 Nilai NRlow

n	$Dlow$(n)	Mean Dlow(n)	NRlow(n)
0	-0.0192	-0.1353	0.1161
1	-0.0544	-0.1353	0.0809
2	-0.0231	-0.1353	0.1122
…	…	…	…
13782	-0.1383	-0.1353	-0.0030

Untuk nilai dari $Dhigh(n)$ maka nilai NRhigh-nya ditunjukkan pada tabel 3.26, dimana NRhigh[n] = Dhigh[n] – E Dhigh[n]

Tabel 3. 26 Nilai NRhigh

n	$Dhigh$(n)	Mean Dhigh(n)	NRhigh(n)
0	-0.0033	-0.0109	0.0076
1	-0.0017	-0.0109	0.0092
2	-0.0001	-0.0109	0.0108
…	…	…	
13781	-0.351	-0.0109	-0.0242

6. *Autocorrelation*

Setelah mendapatkan nilai dari proses *mean removal* yang berupa $NRlow$ dan $NRhigh$, proses selanjutnya yang akan dilakukan adalah perhitungan *autocorrelation*. Pada proses ini nilai NRlow dan NRhigh digabungkan terlebih dahulu menjadi NR seperti ditunjukkan pada tabel 3.27. Setelah digabungkan, nilai *autocorrelation* dihitung dengan persamaan (2.11):

$$AR[k] = \frac{\sum_{n=k}^{N-(k+1)} (NR[n])(NR[n+k])}{\sum_{n=0}^{N-1} (NR[n])^2}$$

Dimana:

k=2205

Tabel 3. 27 Nilai Pembagi *Autocorrelation*

n	NR[n]	$(NR[n])^2$
0	0.1161	0.0135
1	0.0809	0.0065
2	0.1122	0.0126
…	…	…
27563	-0.0242	0.0006
Jumlah		223.8348

Untuk k=1

$$AR[1] = \frac{\sum_{n=1}^{27564-(2)}(NR[n])(NR[n+k])}{\sum_{n=0}^{N-1}(NR[n])^2} = \frac{\sum_{n=1}^{27562}(NR[n])(NR[n+k])}{\sum_{n=0}^{N-1}(NR[n])^2}$$

$$AR[1] = \frac{((NR[1])\,(NR[2]) + (NR[2])\,(NR[3] + \ldots + (NR[27562])\,(NR[27563]))}{223.8348}$$

$$= \frac{((0.0809)\,(0.1122) + (0.1122)\,(0.1302) + \ldots + (0.0079)\,(-0.0242))}{223.8348}$$

$$= 0.5725$$

Akan dilakukan perhitungan yang sama untuk nilai *autocorrelation* yang lain sehingga didapatkan nilainya seperti yang ditunjukkan pada tabel 3.28.

Tabel 3. 28 Hasil Nilai *Autocorrelation*

k	AR[k]
1	0.5725
2	0.5213
3	0.5085
…	…
2205	0.2416

Setelah didapatkan nilai *autocorrelation*-nya, akan diambil 4 nilai tertingginya sebagai nilai keluaran dari fitur ritme yaitu peak1, peak2, peak3 dan peak4 seperti yang ditunjukkan pada tabel 3.29.

Tabel 3. 29 Nilai Keluaran Fitur Ritme

Nilai Fitur Ritme	$Autocorreleation$
Peak1	0.5725
Peak2	0.5331
Peak3	0.5264
Peak4	0.5245

3.3.2 Analisis Pelatihan

Tahapan selanjutnya adalah pelatihan dengan menggunakan *Naive Bayes Classifier*, pelatihan dalam naïve bayes dilakukan untuk mendapatkan data pelatihan yang diestimasi dengan nilai mean dan varians. Nilai mean dan varians

ini akan menjadi acuan dalam melakukan testing. Langkah – langkah pada tahap pelatihan ini adalah mencari nilai mean dan varians dari setiap fitur pada setiap kelas data latih. Nilai fitur permukaan musik dan fitur ritme pada kelas blues akan ditunjukkan pada tabel 3.30 dan tabel 3.31.

Tabel 3. 30 Nilai Fitur Permukaan Musik Pada Kelas Blues

Nama Musik	Mean-Cent	Std-Cent	Mean Roll	Std-Roll	Mean-Flux	Std-Flux	Mean-Zero	Std-Zero	Low Energ
Blues1	218.652	1.454	0.983	0.008	2621.881	3563.838	37.19	16.23	0.592
Blues2	216.566	3.982	0.992	0.008	1601.400	2240.302	20.05	18.05	0.576
Blues3	219.009	1.409	0.982	0.024	3438.596	2995.509	32.09	26.64	0.488
…	…	…	…	…	…	…	…	…	…
Blues90	216.9760	2.5122	0.9972	0.003	1165.4170	2988.6270	9.5040	5.4877	0.6200

Tabel 3. 31 Nilai Fitur Ritme pada kelas Blues

Nama Musik	1^{st} Peak	2^{nd} Peak	3^{rd} Peak	4^{th} Peak
Blues1	0.5725	0.5331	0.5264	0.5245
Blues2	0.7594	0.6401	0.5941	0.5845
Blues3	0.5876	0.5540	0.5384	0.5317
…				
Blues90	0.9218	0.8382	0.8355	0.8180

Dari *dataset* diatas, kemudian dihitung *mean* dan *varians* setiap kelasnya, fungsinya adalah untuk mendapatkan nilai *mean* dan *varian* yang nantinya akan digunakan untuk pencarian probabillitas *likehood*, dengan rumus densitas.

1. menghitung rata-rata (mean) tiap fitur pada tiap kelas dengan persamaan di bawah ini.

$\mu = \frac{\sum x_i}{n}$

Dimana:

μ = *mean*

n = banyaknya data

$\sum x_i$ = jumlah nilai data

2. Kemudian hitung nilai *Varians* tiap fitur pada tiap kelas dengan persamaan di bawah ini.

$\sigma^2 = \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

Dimana:

σ^2= *varians*

μ= *mean*

xi = nilai data

n = banyaknya data

Mean-Cent

μ *mean-cent blues* $= \frac{218.6529+216.5665+219.0091+\cdots+216.9760}{90} = 217.8663$

σ^2 *mean-cent blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((218.6529 - 217.8663)^2 + (216.5665 - 217.8663)^2 +$

$(219.0091 - 217.8663)^2 + \ldots + (216.9760 - 217.8663)^2)$

$= 2.5050$

Std-Cent

μ *std-cent blues* $= \frac{1.4543+3.9820+1.4090+\cdots+2.5122}{90} = 2.4810$

σ^2 *std-cent blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((1.4543 - 2.4810)^2 + (3.9820 - 2.4810)^2 +$

$(1.4090 - 2.4810)^2 + \ldots + (2.5122 - 2.4810)^2)$

$= 2.7085$

Mean-Rolloff

μ *mean-rolloff* blues $= \frac{0.9836+0.9926+0.9823+\cdots+0.9972}{90} = 0.9837$

σ^2 *mean-rolloff blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((0.9836 - 0.9837)^2 + (0.9926 - 0.9837)^2 +$

$(0.9823 - 0.9837)^2 + \ldots + (0.9972 - 0.9837)^2)$

$= 0.00006$

Std-Rolloff

μ *std-rolloff blues* $= \frac{0.0084+0.0089+0.0246+\cdots+0.0034}{90} = 0.0134$

σ^2 *std-rolloff blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \mu)^2$

$= \frac{1}{(90-1)}((0.0084 - 0.0134)^2 + (0.0089 - 0.0134)^2 +$

$(0.0246 - 0.0134)^2 + \ldots + (0.0034 - 0.0134)^2)$

$= 0.00006$

Mean-Flux

μ *mean-flux* blues $= \frac{2621.8817+1601.4003+3438.5962+\cdots+1165.4170}{90} = 3249.3368$

σ^2 *mean-flux blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \mu)^2$

$= \frac{1}{(90-1)}((2621.8817 - 3249.3368)^2 + (1601.4003 - 3249.3368)^2 +$

$(3438.5962 - 3249.3368)^2 + \ldots + (1165.4170 - 3249.3368)^2)$

$= 5195592.1861$

Std-Flux

μ *std-flux blues* $= \frac{3563.8382+2240.3021+2995.5092+\cdots+2988.6270}{90} = 3340.7215$

σ^2 *std-flux blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \mu)^2$

$= \frac{1}{(90-1)}((3563.8382 - 3340.7215)^2 + (2240.3021 - 3340.7215)^2 +$

$(2995.5092 - 3340.7215)^2 + \ldots + (2988.6270 - 3340.7215)^2)$

$= 4135022.8209$

Mean-Zerocrossings

μ *mean-zero* blues $= \frac{37.1960+20.0560+32.0960+\cdots+9.5040}{90} = 36.1789$

σ^2 *mean-zero blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \mu)^2$

$$= \frac{1}{(90-1)}((37.1960 - 36.1789\ )^2 + (20.0560 - \ 36.1789)^2 \quad +$$

$$(32.0960 - \ 36.1789)^2 + \ldots + (9.5040 - \ 36.1789\ )^2)$$

$$= \quad 276.9389$$

Std-Zerocrossings

μ *std-zero blues* $= \frac{16.2302+18.0500+26.6449+\cdots+5.4877}{90} = 19.6514$

σ^2 *std-zero blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \ \mu\ )^2$

$$= \frac{1}{(90-1)}((16.2302 - 19.6514\ )^2 + (18.0500 - \ 19.6514\ )^2 +$$

$$(26.6449 - 19.6514\ )^2 + \ldots + (5.4877 - 19.6514\ )^2)$$

$$= \quad 60.7060$$

LowEnergy

μ *LowEnergy* blues $= \frac{0.5920+0.5760+0.4880+\cdots+0.6200}{90} = 0.5687$

σ^2 *LowEnergy blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \ \mu\ )^2$

$$= \frac{1}{(90-1)}((0.5920 - \ 0.5687\ )^2 + (0.5760 - 0.5687\ )^2 +$$

$$(\ 0.4880 - 0.5687\ )^2 + \ldots + (0.6200 - \ 0.5687)^2)$$

$$= \quad 0.0032$$

P**eak1**

μ *peak 1* blues $= \frac{0.5725+0.7594+0.5876+\cdots+0.9218}{90} = 0.6775$

σ^2 *peak 1 blues* $= \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \ \mu\ )^2$

$$= \frac{1}{(90-1)}((0.5725 - \ 0.6775\ )^2 + (0.7594 - \ 0.6775\ )^2 +$$

$$(\ 0.5876 - \ 0.6775)^2 + \ldots + (0.9218 - \ 0.6775)^2)$$

$$= \quad 0.0121$$

Peak2

μ *peak 2 blues* $= \frac{0.5331+0.6401+0.5540+\cdots+0.8382}{90} = 0.5953$

σ^2 *peak 2 blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((0.5331 - 0.5953)^2 + (0.6401 - 0.5953)^2 +$

$(0.5540 - 0.5953)^2 + \ldots + (0.8382 - 0.5953)^2)$

$= 0.0091$

Peak3

μ *peak 3* blues $= \frac{0.5264+0.5941+0.5384+\cdots+0.8355}{90} = 0.5699$

σ^2 *peak 3 blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((0.5264 - 0.5699)^2 + (0.5941 - 0.5699)^2 +$

$(0.5384 - 0.5699)^2 + \ldots + (0.8355 - 0.5699)^2)$

$= 0.0084$

Peak4

μ *peak 4 blues* $= \frac{0.5245+0.5845+0.5317+\cdots+0.8180}{90} = 0.5583$

σ^2 *peak 4 blues* $= \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \mu)^2$

$= \frac{1}{(90-1)} ((0.5245 - 0.5583)^2 + (0.5845 - 0.5583)^2 +$

$(0.5317 - 0.5583)^2 + \ldots + (0.8180 - 0.5583)^2)$

$= 0.0076$

Setelah didapatkan nilai mean dan varian dari fitur permukaan musik dan fitur ritme pada kelas blues, maka akan dilakukan hal yang sama untuk 9 kelas lainnya seehingga akan didapatkan nilai mean dan varian seperti yang ditunjukkan pada tabel 3.32.

Tabel 3. 32 Nilai Mean dan Varian dari 10 kelas

	Kelas	Mean-Cent	Std-Cent	Mean-Roll	Std-Roll	Mean-Flux	Std-Flux	Mean Zero
Mean	Blues	217.86	2.481	0.9837	0.0134	3249.3368	3340.7215	36.1789
	Classic	217.74	2.613	0.9824	0.0069	667.0245	584.3490	31.9611
	Countr	218.58	1.738	0.9838	0.0128	2157.3899	1762.7633	35.4332
	Disco	218.43	1.945	0.9711	0.0227	2668.7215	3088.8684	55.8558
	Hiphop	217.88	2.718	0.9747	0.0281	5903.8487	7647.3353	47.5953
	Jazz	218.06	2.261	0.9815	0.0120	1844.9888	1634.9266	35.1572
	Metal	218.85	1.440	0.9647	0.0175	3113.0420	2326.3320	66.4884
	Pop	218.40	2.284	0.9689	0.0334	5839.4854	6774.3825	57.2354
	Reggae	217.15	3.315	0.9803	0.0232	2381.2784	2742.2494	38.1384
	Rock	218.61	1.707	0.9762	0.0174	2435.4429	2019.9702	48.8736
Var	Blues	2.5050	2.708	0.0006	0.0006	5195592.186	4135022.820	276.938
	Classic	6.5688	8.862	0.0001	0.0001	1762657.161	862610.7401	128.163
	Country	0.6534	0.487	0.0001	0.0001	4047497.099	1192661.649	208.748
	Disco	2.4830	1.340	0.0001	0.0001	3700396.388	4567320.859	256.130
	Hiphop	1.0814	1.188	0.0001	0.0002	10383678.22	24592088.27	264.721
	Jazz	2.5922	2.025	0.0001	0.0001	3605755.852	2344119.162	366.352
	Metal	0.3735	0.480	0.0001	0.0001	7044457.471	4165799.560	237.540
	Pop	0.7213	0.998	0.0001	0.0003	12596572.25	28192176.07	334.676
	Reggae	3.2645	4.206	0.0001	0.0001	5382920.081	6581306.129	261.981
	Rock	1.1504	1.156	0.0001	0.0001	2826021.859	1589669.920	331.988

	Kelas	Std Zero	Low Energy	1^{st} Peak	2^{nd} Peak	3^{rd} Peak	4^{th} Peak
Mean	Blues	19.651	0.5687	0.677	0.595	0.569	0.558
	Classic	10.630	0.5657	0.658	0.623	0.607	0.596
	Countr	19.182	0.5561	0.658	0.575	0.553	0.542
	Disco	30.773	0.5685	0.662	0.593	0.554	0.533
	Hiphop	34.689	0.5746	0.777	0.694	0.639	0.606
	Jazz	16.183	0.5471	0.634	0.587	0.571	0.559
	Metal	21.989	0.5568	0.579	0.496	0.460	0.444
	Pop	39.943	0.5817	0.695	0.616	0.580	0.553
	Reggae	30.166	0.5796	0.752	0.677	0.630	0.599
	Rock	24.841	0.5503	0.613	0.539	0.509	0.496
Var	Blues	60.706	0.0032	0.012	0.009	0.008	0.007
	Classic	28.787	0.0057	0.009	0.007	0.007	0.006
	Country	98.429	0.0032	0.007	0.005	0.005	0.005
	Disco	140.03	0.0037	0.010	0.007	0.005	0.003
	Hiphop	169.20	0.0032	0.009	0.009	0.007	0.006
	Jazz	84.002	0.0036	0.010	0.008	0.007	0.006
	Metal	44.605	0.0032	0.012	0.008	0.005	0.004
	Pop	283.12	0.0032	0.012	0.0099	0.0075	0.0057
	Reggae	158.31	0.0028	0.009	0.0082	0.0058	0.0039
	Rock	78.494	0.0030	0.013	0.0067	0.0043	0.0033

3.3.3 Analisis Pengujian

Setelah didapatkan nilai mean dan varian dari data training pada tahap pelatihan, maka selanjutnya bisa dilakukan testing terhadap data baru, langkah-langkahnya adalah sebagai berikut:

1. Mencari nilai fitur permukaan musik dan fitur ritme.

Misal data testing yang akan digunakan adalah classical1, akan dilakukan perhitungan nilai fitur permukaan musik dan fitur ritme sehingga dihasilkan nilai seperti ditunjukkan pada tabel 3.33

Tabel 3. 33 Nilai Ekstraksi Fitur Data Testing Classical91

Kelas	Nama Musik	Mean-Cent	Std-Cent	Mean-Rolloff	Std-Rolloff	Mean-Flux	Std-Flux	Mean Zero
????	Classical91	216.6724	2.8451	0.9582	0.0186	154.0585	228.1503	67.0600
		Std Zero	Low Energy	1^{st} Peak	2^{nd} Peak	3^{rd} Peak	4^{th} Peak	
		24.3013	0.6200	0.7114	0.7110	0.6455	0.6232	

2. Mencari prior masing-masing kelas pada data training.

Prior blues = 90/900 = 0.1

Prior classical = 90/900 = 0.1

Prior country = 90/900 = 0.1

Prior disco = 90/900 = 0.1

Prior hiphop = 90/900 = 0.1

Prior jazz = 90/900 = 0.1

Prior metal = 90/900 = 0.1

Prior pop = 90/900 = 0.1

Prior reggae = 90/900 = 0.1

Prior rock = 90/900 = 0.1

3. Mencari densitas probabilitas data testing.

Densitas probabilitas data testing akan dihitung dengan distribusi normal seperti pada persamaan (2.15).

$$\varphi_{\mu\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.15)$$

Dimana :

x = data masukan

π = 3,14

σ = standar deviasi

σ^2 = varian

μ = mean

Densitas Probabilitas Mean-Centroid kelas Blues

$$\varphi \text{ (mean-centroid|Blues)} = \frac{1}{\sqrt{2\pi\,(2.5050)}} e^{-\frac{(216.6724-217.8663)^2}{2\,(2.5050)}}$$

$$= \frac{1}{\sqrt{15.7314}} e^{-\frac{(1.4254)}{5.01}} = \frac{1}{3.9663} e^{-(0.2845)}$$

$$= 0.1896$$

Densitas Probabilitas Std-Centroid kelas Blues

$$\varphi \text{ (std-centroid|Blues)} = \frac{1}{\sqrt{2\pi\,(2.7085)}} e^{-\frac{(2.8451-2.4810)^2}{2\,(2.7085)}}$$

$$= \frac{1}{\sqrt{17.0094}} e^{-\frac{(0.1326)}{5.417}} = \frac{1}{4.1242} e^{-(0.0245)}$$

$$= 0.2365$$

Densitas Probabilitas Mean-Rolloff kelas Blues

$$\varphi \text{ (mean-rolloff|Blues)} = \frac{1}{\sqrt{2\pi\,(0.00006)}} e^{-\frac{(0.9582-0.9837)^2}{2\,(0.00006)}}$$

$$= \frac{1}{\sqrt{0.0003768}} e^{-\frac{(0.00065025)}{0.00012}} = \frac{1}{0.0194} e^{-(5.41875)}$$

$$= 0.2908$$

Densitas Probabilitas Std-Rolloff kelas Blues

$$\varphi \text{ (std-rolloff|Blues)} = \frac{1}{\sqrt{2\pi\,(0.00006)}} e^{-\frac{(0.0186-0.0134)^2}{2\,(0.00006)}}$$

$$= \frac{1}{\sqrt{0.0003768}} e^{-\frac{(0.0052)}{0.00012}} = \frac{1}{0.0194} e^{-(43.3333)}$$

$$= \quad 40.9772$$

Densitas Probabilitas Mean-Flux kelas Blues

$$\varphi \text{ (mean-flux|Blues)} = \frac{1}{\sqrt{2\pi\,(5195592.1861)}}\, e^{-\frac{(154.0585-3249.3368)^2}{2\,(5195592.1861)}}$$

$$= \quad 0.0026$$

Densitas Probabilitas Std-Flux kelas Blues

$$\varphi \text{ (std-flux|Blues)} = \frac{1}{\sqrt{2\pi\,(4135022.8209\,)}}\, e^{-\frac{(228.1503-3340.7215)^2}{2\,(4135022.8209)}}$$

$$= \quad 0.0051$$

Densitas Probabilitas Mean-Zerocrossings kelas Blues

$$\varphi \text{ (mean-zerocrossings|Blues)} = \frac{1}{\sqrt{2\pi\,(276.9389)}}\, e^{-\frac{(67.0600-36.1789)^2}{2\,(276.9389)}}$$

$$= \frac{1}{\sqrt{1739.1763}}\, e^{-\frac{(953.6423)}{553.8778}} = \frac{1}{41.7034}\, e^{-(1.7217)}$$

$$= \quad 0.0043$$

Densitas Probabilitas Std-Zerocrossings kelas Blues

$$\varphi \text{ (std-zerocrossings|Blues)} = \frac{1}{\sqrt{2\pi\,(60.7060)}}\, e^{-\frac{(24.3013-19.6514)^2}{2\,(60.7060)}}$$

$$= \frac{1}{\sqrt{381.2337}}\, e^{-\frac{(21.6216)}{121.412}} = \frac{1}{19.5252}\, e^{-(0.1781)}$$

$$= \quad 0.0429$$

Densitas Probabilitas Low Energy kelas Blues

$$\varphi \text{ (low energy|Blues)} = \frac{1}{\sqrt{2\pi\,(0.0032)}}\, e^{-\frac{(0.6200-0.5687)^2}{2\,(0.0032)}}$$

$$= \frac{1}{\sqrt{0.0201}}\, e^{-\frac{(0.002632)}{0.0064}} = \frac{1}{0.1418}\, e^{-(0.4112)}$$

$$= \quad 4.6689$$

Densitas Probabilitas Peak1 kelas Blues

$$\varphi(\text{peak1}|\text{Blues}) = \frac{1}{\sqrt{2\pi\,(0.0121)}} e^{-\frac{(0.7114-0.6775)^2}{2\,(0.0121)}}$$

$$= \frac{1}{\sqrt{0.0760}} e^{-\frac{(0.00115)}{0.0242}} = \frac{1}{0.2757} e^{-(0.0475)}$$

$$= 3.4566$$

Densitas Probabilitas Peak2 kelas Blues

$$\varphi(\text{peak2}|\text{Blues}) = \frac{1}{\sqrt{2\pi\,(0.0091)}} e^{-\frac{(0.7110-0.5953)^2}{2\,(0.0091)}}$$

$$= \frac{1}{\sqrt{0.0571}} e^{-\frac{(0.0134)}{0.0182}} = \frac{1}{0.2389} e^{-(0.6972)}$$

$$= 2.0060$$

Densitas Probabilitas Peak3 kelas Blues

$$\varphi(\text{peak3}|\text{Blues}) = \frac{1}{\sqrt{2\pi\,(0.0084)}} e^{-\frac{(0.6455-0.5699)^2}{2\,(0.0084)}}$$

$$= \frac{1}{\sqrt{0.0527}} e^{-\frac{(0.005715)}{0.0168}} = \frac{1}{0.2296} e^{-(0.3402)}$$

$$= 3.0986$$

Densitas Probabilitas Peak4 kelas Blues

$$\varphi(\text{peak4}|\text{Blues}) = \frac{1}{\sqrt{2\pi\,(0.0076)}} e^{-\frac{(0.6232-0.5583)^2}{2\,(0.0076)}}$$

$$= \frac{1}{\sqrt{0.0477}} e^{-\frac{(0.0649)}{0.0152}} = \frac{1}{0.2184} e^{-(4.2697)}$$

$$= 3.4723$$

Setelah didapatkan nilai densitas probabilitas tiap fitur pada kelas blues, selanjutnya akan dilakukan perhitungan terhadap 9 kelas yang lain dan hasilnya ditunjukkan pada tabel 3.34.

Tabel 3. 34 Hasil Perhitungan Densitas Probabilitas Pada Tiap Kelas

Kelas	Mean-Cent	Std-Cent	Mean-Roll	Std-Roll	Mean-Flux	Std-Flux	Mean Zero	Std Zero
Blues	0.1896	0.236	0.2908	40.977	0.00007	0.00006	0.0043	0.0429
Classic	0.1425	0.133	0.0358	0.4294	0.00028	0.00040	0.0003	0.0029

Countr	0.0301	0.162	0.4205	32.802	0.00012	0.00014	0.0025	0.0352
Disco	0.1353	0.254	18.436	27.828	0.00009	0.00008	0.0195	0.0290
Hiphop	0.1946	0.363	13.465	22.359	0.00003	0.00003	0.0120	0.0223
Jazz	0.1703	0.257	4.5400	34.950	0.00014	0.00017	0.0052	0.0294
Metal	0.0011	0.073	26.664	55.357	0.00008	0.00012	0.0259	0.0563
Pop	0.0587	0.341	21.573	15.700	0.00003	0.00004	0.0189	0.0154
Reggae	0.2129	0.189	7.3180	27.779	0.00011	0.00010	0.0050	0.0284
Rock	0.0718	0.211	11.017	44.767	0.00009	0.00012	0.0133	0.0449

Kelas	Low Energy	1^{st} Peak	2^{nd} Peak	3^{rd} Peak	4^{th} Peak
Blues	4.6689	3.4566	2.0060	3.0986	3.4723
Classic	4.0794	3.5108	2.7622	4.3011	4.6849
Countr	3.7272	3.9220	1.0391	2.5028	2.9690
Disco	4.5855	3.5015	1.8394	2.4958	2.1799
Hiphop	5.1167	3.2101	4.0002	4.5321	4.8134
Jazz	3.1801	2.9465	1.7364	3.1994	3.6238
Metal	3.7752	1.7919	0.2529	0.1869	0.1282
Pop	5.5908	3.5654	2.5419	3.4774	3.4443
Reggae	5.6115	3.7095	4.1176	5.1474	5.9712
Rock	3.2324	2.4032	0.5442	0.7002	0.6134

4. Setelah didapatkan nilai densitas probabilitasnya, selanjutnya menghitung evidence. Evidence menghitung jumlah dari prior dikali dengan densitas probabilitas masing-masing kelas.

Evidence = (P(blues) . P(mean-cent|blues) . P(std-cent|blues) .
P(mean-rolloff|blues) . P(std-rolloff|blues) . P(mean-flux|blues) .
P(std-flux|blues) . P(mean-zero|blues) . P(std-zero|blues) .
P(lowenergry|blues) . P(peak1|blues) . P(peak2|blues) .
P(peak3|blues) . P(peak4|blues)) + … + (P(rock) . P(mean-cent| rock) .
P(std-cent| rock) . P(mean-rolloff| rock) . P(std-rolloff| rock) .
P(mean-flux| rock) . P(std-flux| rock) . P(mean-zero| rock) .
P(std-zero| rock) . P(lowenergry| rock) . P(peak1| rock) .
P(peak2| rock) . P(peak3| rock) . P(peak4| rock))

Evidence = ((0.1 * 0.1896 * 0.2365 * 0.2908 * 40.9772 * 0.00007 * 0.00006 *
0.0043*0.0429*4.6689*3.4566*2.0060*3.0986*3.4723) + … +
(0.1 * 0.0718 * 0.2119 * 11.0177 * 44.7672 * 0.00009 * 0.00012 *
0.0133 * 0.0449 * 3.2324 * 2.4032 * 0.5442 * 0.7002 * 0.6134))

= 0.000000005

5. Setelah didapatkan nilai evidence, selanjutnya posterior masing-masing kelas dihitung dengan menggunakan persamaan (2.16).

$$Posterior = \frac{P(Ci) \text{x } P(X1 \ldots\ldots\ldots\ldots\ldots\ldots Xn|Ci)}{P(X1 \ldots\ldots\ldots\ldots\ldots\ldots Xn|Ci)}$$

atau bisa ditulis

$$posterior = \frac{\text{prior x likehood}}{\text{evidence}} \qquad (2.16)$$

Posterior(Blues) = (0.1 * 0.1896 * 0.2365 * 0.2908 * 40.9772 * 0.00007 * 0.00006* 0.0043 * 0.0429 * 4.6689 * 3.4566 * 2.0060 * 3.0986 * 3.4723) / 0.000000005

= 0.002648

Posterior(Classical) = (0.1 * 0.1425 * 0.1336 * 0.0358 * 0.4294 * 0.00028 * 0.00040 * 0.0003 * 0.0029 * 4.0794 * 3.5108 * 2.7622 * 4.3011 * 4.6849) / 0.000000005

= 0.0000004

Posterior(Country) = (0.1 * 0.0301 * 0.1627 * 0.4205 * 32.8020 * 0.00012 * 0.00014 * 0.0025 * 0.0352 * 3.7272 * 3.9220 * 1.0391 * 2.5028 * 2.9690) / 0.000000005

= 0.000203

Posterior (Disco) = (0.1 * 0.1353 * 0.2548 * 18.4361 * 27.8280 * 0.00009 * 0.00008 * 0.0195 * 0.0290 * 4.5855 * 3.5015 * 1.8394 * 2.4958 * 2.1799) / 0.000000005

= 0.198072

Posterior (Hiphop) = (0.1 * 0.1946 * 0.3634 * 13.4653 * 22.3591 * 0.00003 *

0.00003 * 0.0120 * 0.0223 * 5.1167 * 3.2101 * 4.0002 * 4.5321 * 4.8134) / 0.000000005

= 0.098808

Posterior (Jazz) = (0.1 * 0.1703 * 0.2577 * 4.5400 * 34.9509 * 0.00014 * 0.00017 * 0.0052 * 0.0294 * 3.1801 * 2.9465 * 1.7364 * 3.1994 * 3.6238) / 0.000000005

= 0.088656

Posterior (Metal) = (0.1 * 0.0011 * 0.0738 * 26.6641 * 55.3570 * 0.00008 * 0.00012 * 0.0259 * 0.0563 * 3.7752 * 1.7919 * 0.2529 * 0.1869 * 0.1282 / 0.000000005

= 0.000001

Posterior (Pop) = (0.1 * 0.0587 * 0.3410 * 21.5733 * 15.7003 * 0.00003 * 0.00004 * 0.0189 * 0.0154 * 5.5908 * 3.5654 * 2.5419 * 3.4774 * 3.4443 / 0.000000005

= 0.023933

Posterior (Reggae) = (0.1 * 0.2129 * 0.1895 * 7.3180 * 27.7792 * 0.00011 * 0.00010 * 0.0050 * 0.0284 * 5.6115 * 3.7095 * 4.1176 * 5.1474 * 5.9712) / 0.000000005

= 0.586055

Posterior (Rock) = (0.1 * 0.0718 * 0.2119 * 11.0177 * 44.7672 * 0.00009 * 0.00012 * 0.0133 * 0.0449 * 3.2324 * 2.4032 * 0.5442 * 0.7002 * 0.6134) / 0.000000005

= 0.001623

3.3.4 Analisis Data Keluaran

Data keluaran adalah data hasil setelah melakukan serangkaian proses. Proses yang dilakukan antara lain pencarian nilai ekstraksi fitur permukaan musik dan fitur ritme dan dilanjutkan dengan tahap klasifikasi yaitu tahap pelatihan dan pengujian dengan metode naïve bayes. Setelah melalui serangkaian proses tersebut, maka akan dihasilkan data keluaran berupa hasil klasifikasi. Data keluaran didapatkandari hasil klasifikasi yang merupakan nilai posterior terbesar.

Dari analisis sebelumnya setelah dihitung posterior masing-masing kelas, akan dicari nilai posterior terbesar dimana kelas dengan nilai posterior terbesar merupakan kelas yang paling cocok untuk data testing. Tabel posterior tiap kelas dapat dilihat pada tabel 3.35.

Tabel 3. 35 Nilai P*osterior* Data *Testing*

Nama Kelas	Nilai Posterior
Blues	0.002648
Classical	0.0000004
Country	0.000203
Disco	0.198072
Hiphop	0.098808
Jazz	0.088656
Pop	0.000001
Metal	0.023933
Reggae	0.586055
Rock	0.001623

Dari tabel 3.35 diatas dapat kita lihat bahwa nilai posterior yang paling besar adalah reggae. Maka sistem akan menampilan reggae sebagai kelas bagi data testing diatas.

3.4 Analisis Kebutuhan Non-Fungsional

Analisis non-sungsional yang dilakukan dibagi kedalam 3 tahap, yaitu analisis kebutuhan perangkat keras, analisis kebutuhan perangkat lunak dan analisis pengguna.

3.4.1 Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras pada penelitian ini merupakan kebutuhan perangkat keras minimum untuk menjalankan aplikasi. Adapun perangkat keras yang digunakan sebagai berikut:

1. Laptop dengan *processor* Intel Core i3 (*recommended: Intel Core i5 or equivalent*).
2. RAM 2GB (*recommended* : 4 GB(64-bit)).
3. *Harddisk* 10 GB (*recommended* : 20 GB *of free disk space*).

3.4.2 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak pada penelitian ini merupakan *tools* yang digunakan dalam pembangunan aplikasi. Adapun *tools* yang digunakan sebagai berikut :

1. OS Windows 8.1 Pro 64 bit.
2. NetBeans IDE 8.2
3. *Java Development Kit* dan *Java Runtime Environtment.*
4. Matlab R2016a

3.4.3 Analisis Pengguna

Analisis kebutuhan pengguna merupakan uraian mengenai siapa saja yang akan menggunakan sistem dan terlibat dalam pengolahan data beserta karakteristiknya sehingga dapat diketahui tingkat pengalaman dan pemahaman pengguna terhadap sistem. Adapun karakteristik dari pengguna adalah sebagai berikut:

1. Pengguna terbiasa mendengarkan lagu dari berbagai genre.
2. Pengguna minimal mampu menggunakan keyboard dan mouse sebagai alat untuk berinteraksi dengan sistem.
3. Pengguna sudah terbiasa menggunakan komputer.

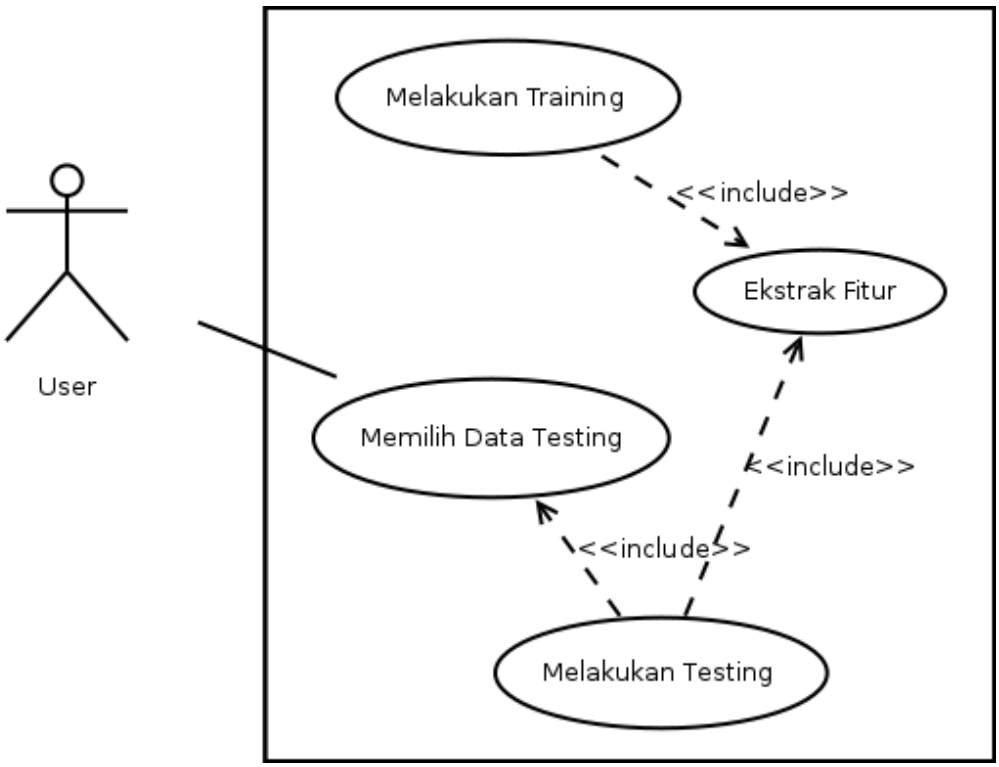
3.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk menganalisis proses yang diterapkan dalam sistem dan menjelaskan kebutuhan yang diperlukan. Menganalisis operasional sistem dengan mendefinisikan skenario penggunaan

aplikasi. Analisis yang dilakukan dimodelkan dengan menggunakan UML (*Unified Modelling Language*). Bagian-bagian yang dilakukan dalam analisis tersebut antara lain *Use Case Diagram, Use Case Scenario, Activity Diagram, Sequence Diagram* dan *Class Diagram.*

3.5.1 *Use Case Diagram*

Use case diagram merupakan pemodelan yang menunjukan interaksi antara sistem dan lingkungannya. Use case diagram memberikan gambaran yang cukup sederhana dari interaksi-interaksi yang terlibat. Berikut adalah perancangan proses-proses pada sistem yang akan dibangun digambarkan dengan use case diagram yang ditunjukkan pada gambar 3.22:



Gambar 3. 22 *Use Case Diagram* Klasifikasi Genre Musik

Berikut ini merupakan deskripsi aktor yang dapat dilihat pada tabel 3.36 dibawah ini.

Tabel 3. 36 Defiinisi Aktor

No	Aktor	Definisi
1	*User*	Aktor adalah *user* yang menjalankan program klasifikasi genre musik dengan *naive bayes*.

Berikut merupakan deskripsi *use case* yang dapat dilihat pada tabel 3.37 dibawah ini.

Tabel 3. 37 Definisi *Use Case*

No	*Use Case*	Definisi
1	Ekstraksi Fitur	Fungsional ini digunakan untuk melakukan ekstraksi fitur data *training* maupun *testing* untuk mendapatkan ciri dari data tersebut.
2	Melakukan *Training*	Fungsional ini digunakan untuk melakukan *training* dengan *naive bayes*, data yang digunakan sudah disimpan di *folder* yang sudah ditentukan.
3	Memilih Data *Testing*	Fungsional ini digunakan *user* untuk memilih data yang akan di *test*.
4	Melakukan *Testing*	Fungsional ini digunakan untuk melakukan *testing* data yang telah dipilih oleh *user*.

3.5.2 *Use Case Scenario*

Bagian ini menjelaskan scenario untuk tiap *use case* yang menggambarkan urutan intraksi actor dengan *use case* tersebut dari awal sampai akhir. *Use case* ekstraksi fitur ditunjukan pada tabel 3.38.

Tabel 3. 38 *Use Case Scenario* Ekstraksi Fitur

Use Case Name	Ekstraksi Fitur	
Related Requirment	*None*	
Goal	Mengekstraksi fitur dari data untuk mendapat ciri dari data tersebut.	
Precondition	Data yang bisa di ekstraksi berformat *.wav*	
Success End Condition	Sistem akan memberikan hasil ekstraksi fitur.	
Failed End Condition	*None*	
Primary Actor	*None*	
Main Flow	*Step*	*Action*
	1	Sistem mendapatkan input lokasi *folder*.
	2	Sistem melakukan ekstraksi fitur pada semua data yang ada di folder yang ditentukan
Extension	*Step*	*Action*
	-	-

Use case melakukan *training* ditunjukkan pada tabel 3.39.

Tabel 3. 39 *Use Case Scenario* Melakukan *Training*

Use Case Name	Melakukan *Training*	
Related Requirment	*Feature extraction*	
Goal	Sistem selesai melakukan *training* pada data yang sudah ditentukan.	
Precondition	Data yang dilatih akan melalui proses ekstraksi fitur terlebih dahulu.	
Success End Condition	Sistem siap digunakan untuk proses *testing*.	
Failed End Condition	*None*	
Primary Actor	*None*	
Main Flow	*Step*	*Action*
	1	Sistem mengambil data fitur yang sudah dihasilkan dari proses ekstraksi fitur.
	2	Sistem melakukan training dengan *naive bayes*.
Extension	*Step*	*Action*
	-	-

Use case memilih data *testing* ditunjukkan pada tabel 3.40.

Tabel 3. 40 *Use Case Scenario* Memilih Data *Testing*

Use Case Name	Memilih Data *Testing*	
Related Requirment	*None*	
Goal	*User* memilih folder yang berisi data *testing.*	
Precondition	Sistem sudah melakukan proses *training*	
Success End Condition	Sistem mendapatkan lokasi folder yang dipilih oleh *user*.	
Failed End Condition	*None*	
Primary Actor	*User*	
Main Flow	*Step*	*Action*
	1	*User* menekan tombol *Choose and Test*.
	2	Sistem menampilkan *file explorer*.
	3	*User* memilih folder yang berisi data *testing*.
Extension	*Step*	*Action*
	-	-

Use case melakukan *testing* ditunjukkan pada tabel 3.41.

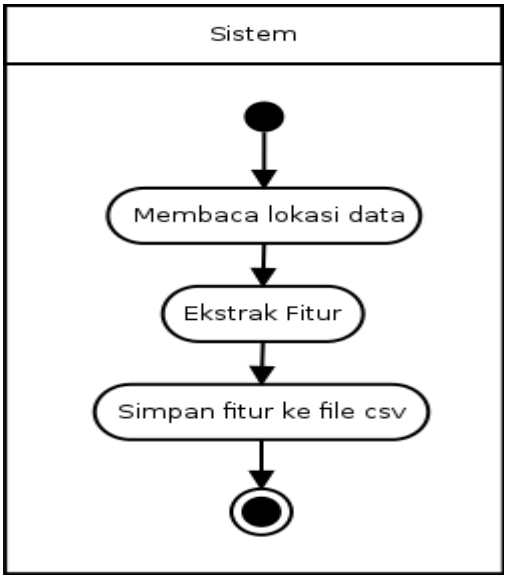
Tabel 3. 41 *Use Case Scenario* Melakukan *Testing*

Use Case Name	Melakukan *Testing*
Related Requirment	*Feature extraction*

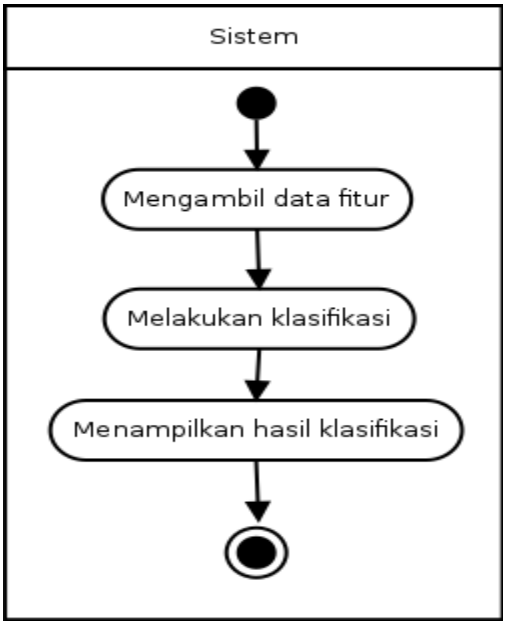
Goal	Sistem selesai melakukan *testing* pada data yang sudah ditentukan.	
Precondition	Data yang dilatih akan melalui proses ekstraksi fitur terlebih dahulu.	
Success End Condition	Sistem menampilkan hasil klasifikasi.	
Failed End Condition	*None*	
Primary Actor	*None*	
Main Flow	*Step*	*Action*
	1	Sistem mengambil data fitur yang sudah dihasilkan dari proses ekstraksi fitur.
	2	Sistem melakukan klasifikasi dengan *naive bayes*.
	3	Sistem menampilkan hasil klasifikasi.
Extension	*Step*	*Action*
	-	-

3.5.3 *Activity Diagram*

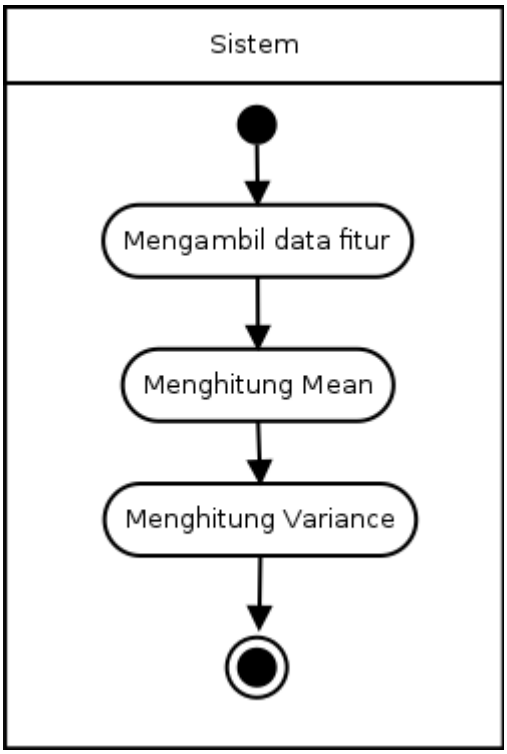
Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum, menggambarkan masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana alir berakhir. Adapun *activity diagram* dari masing-masing *use case scenario* ditunjukkan pada gambar 3.23, gambar3.24, gambar3.25 dan gambar 3.26 sebagai berikut:



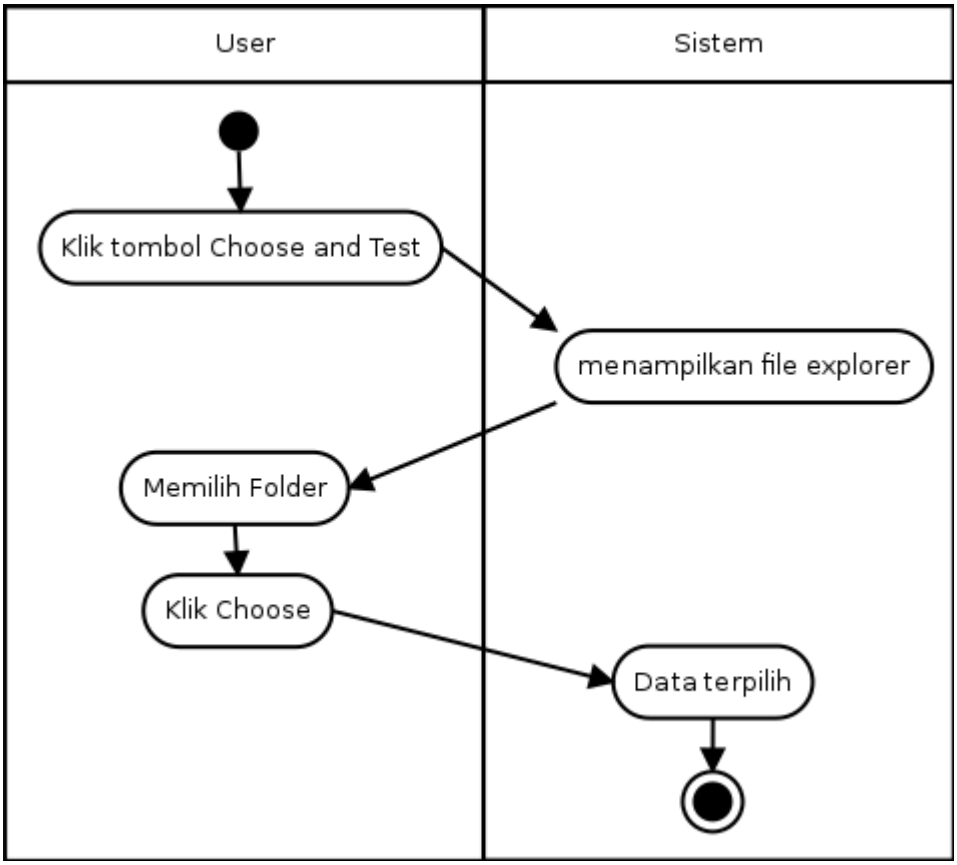
Gambar 3. 23 *Activity Diagram* Ekstraksi Fitur



Gambar 3. 24 *Activity Diagram* Melakukan *Testing*



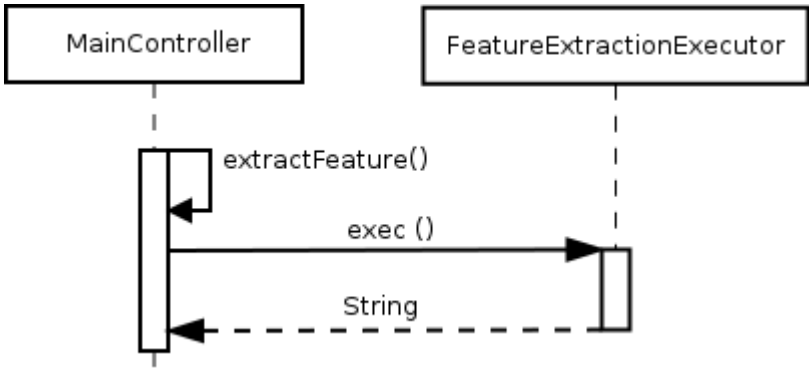
Gambar 3. 25 *Activity Diagram* Melakukan *Training* NBC



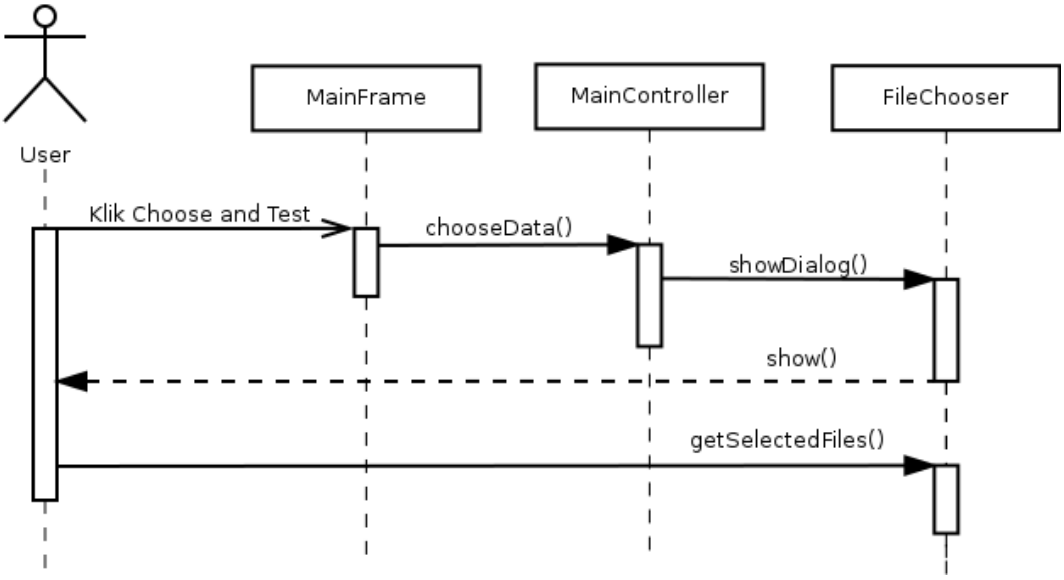
Gambar 3. 26 *Activity Diagram* Memilih Data *Testing*

3.5.4 *Sequence Diagram*

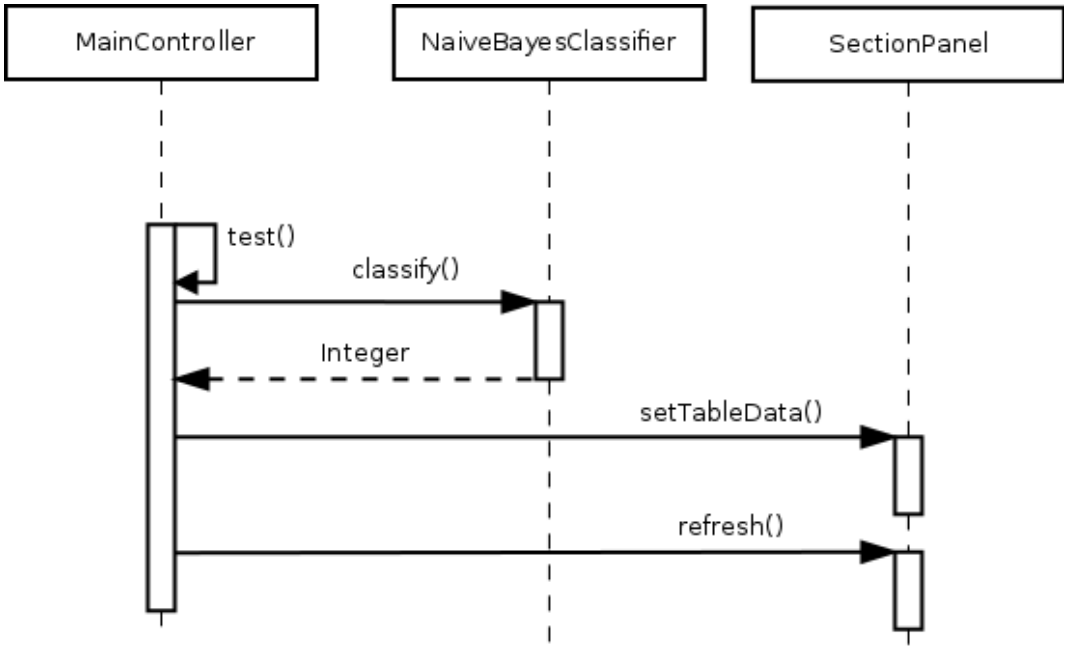
Berikut ini adalah *sequence diagram* yang terdapat pada pembangunan aplikasi yang dibangun. Seperti yang ditunjukkan pada gambar 3.27, gambar 3.28, gambar 3.29 dan gambar 3.30 sebagai berikut:



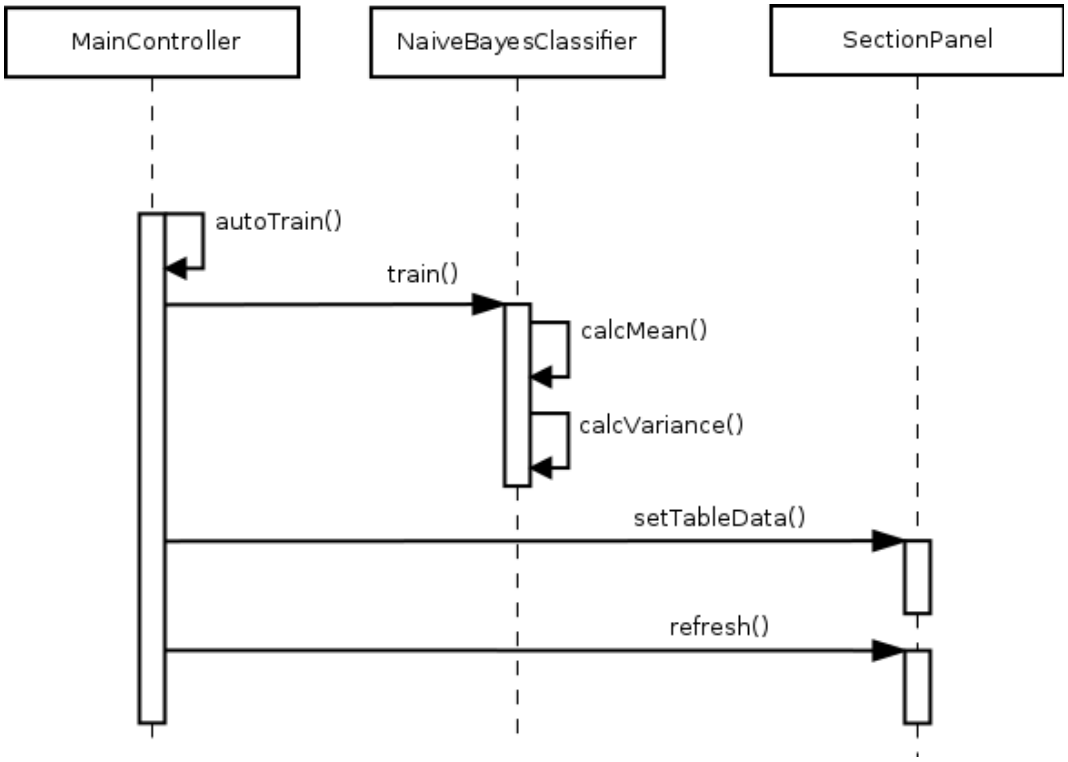
Gambar 3. 27 *Sequence Diagram* Ekstraksi Fitur



Gambar 3. 28 *Sequence Diagram* Memilih Data *Testing*



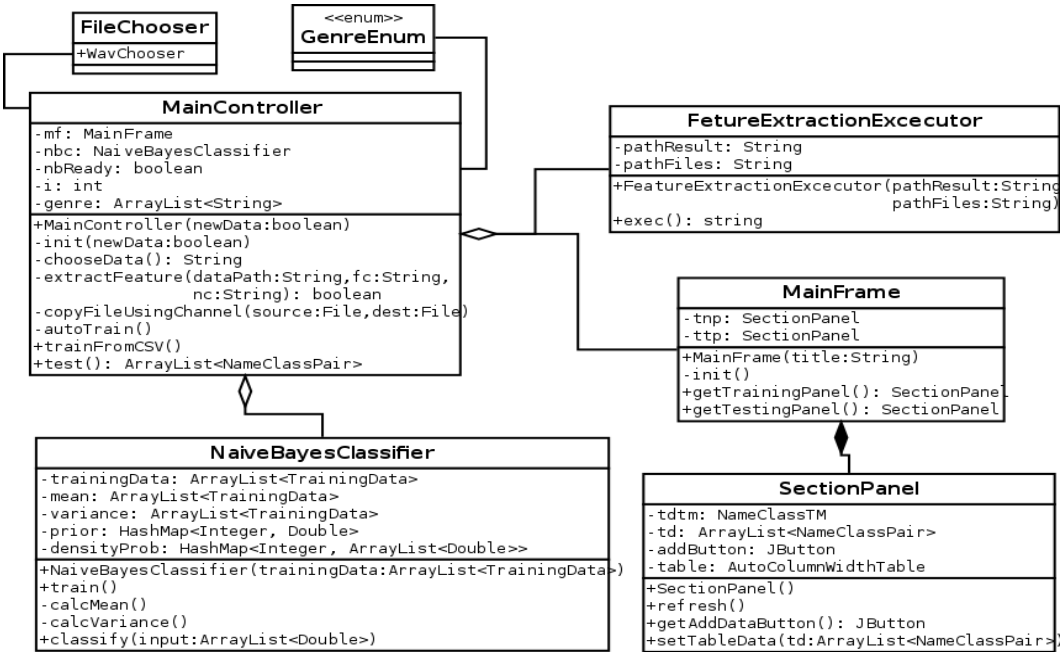
Gambar 3. 29 *Sequence Diagram* Proses *Testing*



Gambar 3. 30 *Sequence Diagram Training* NBC

3.5.5 *Class Diagram*

Class diagram merupakan sebuah spesifikasi yang jika diimajinasikan akan membentuk sebuah objek dan merupakan inti dari desain berorientasi objek. Jenis *Class* dan bentuk *class diagram* pada gambar 3.31 Berikut:



Gambar 3. 31 *Class Diagram*

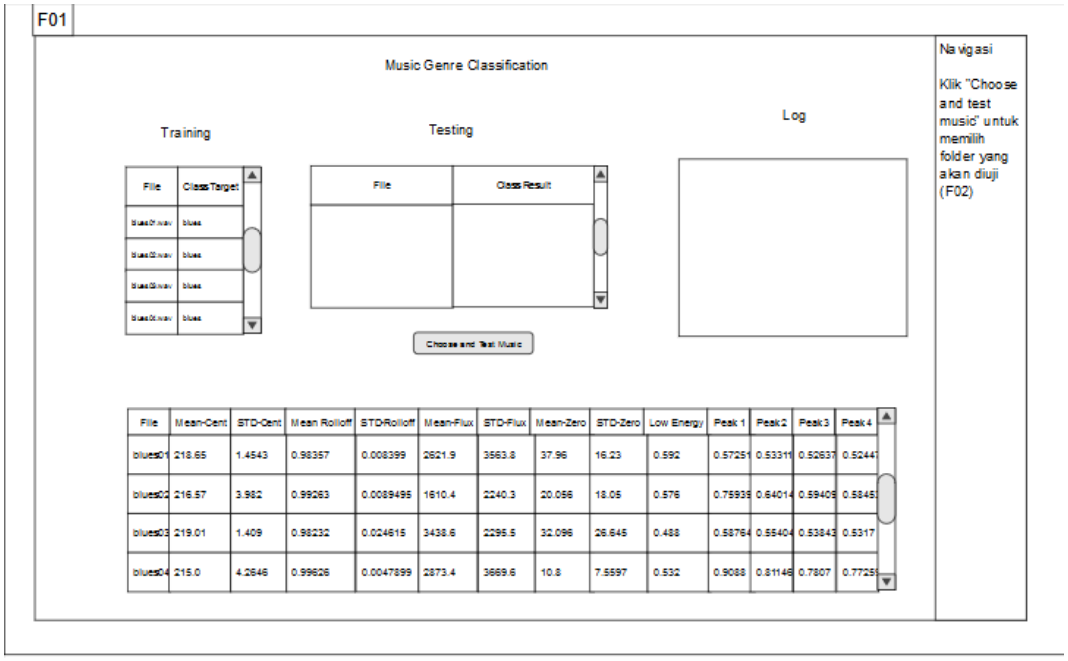
3.6 Perancangan Antarmuka

Perancangan antarmuka menjelaskan rencana tampilan yang akan digunakan pada aplikasi yang akan dibangun. Perancangan antarmuka yang ada terdiri dari perancangan *form* dan jaringan semantik.

3.6.1 Perancangan Form

1. F01 – *Form* Ekstraksi Fitur

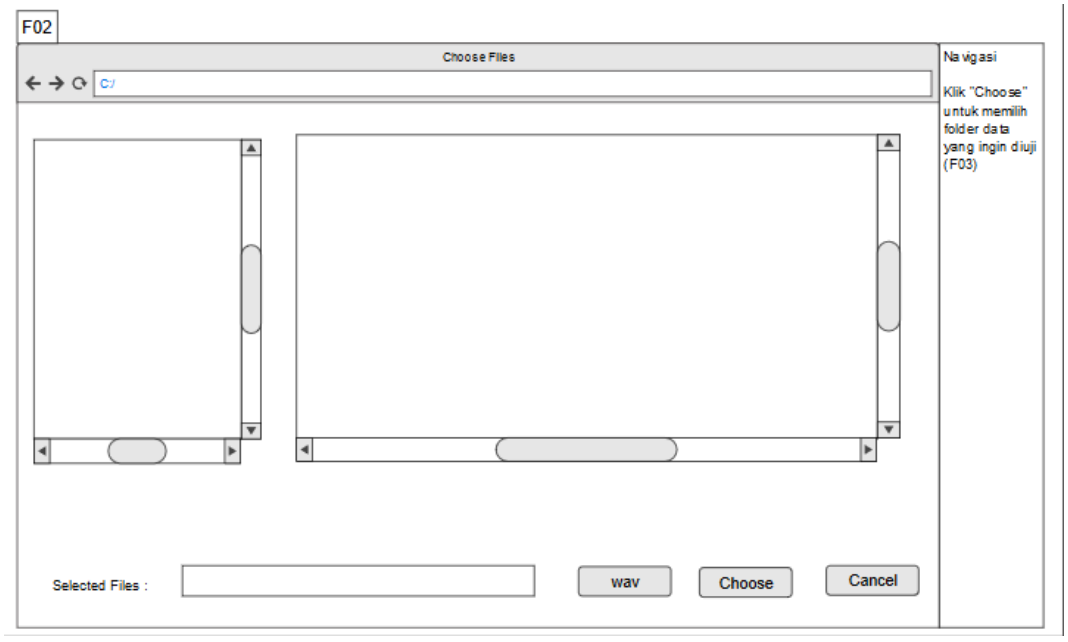
Ketika program dijalankan maka akan otomatis mengekstak fitur data musik yang berada pada folder data *training* seperti yang ditunjukkan pada gambar 3.32



Gambar 3. 32 F01 Ekstraksi Fitur

2. F02 – *Form* memilih data musik

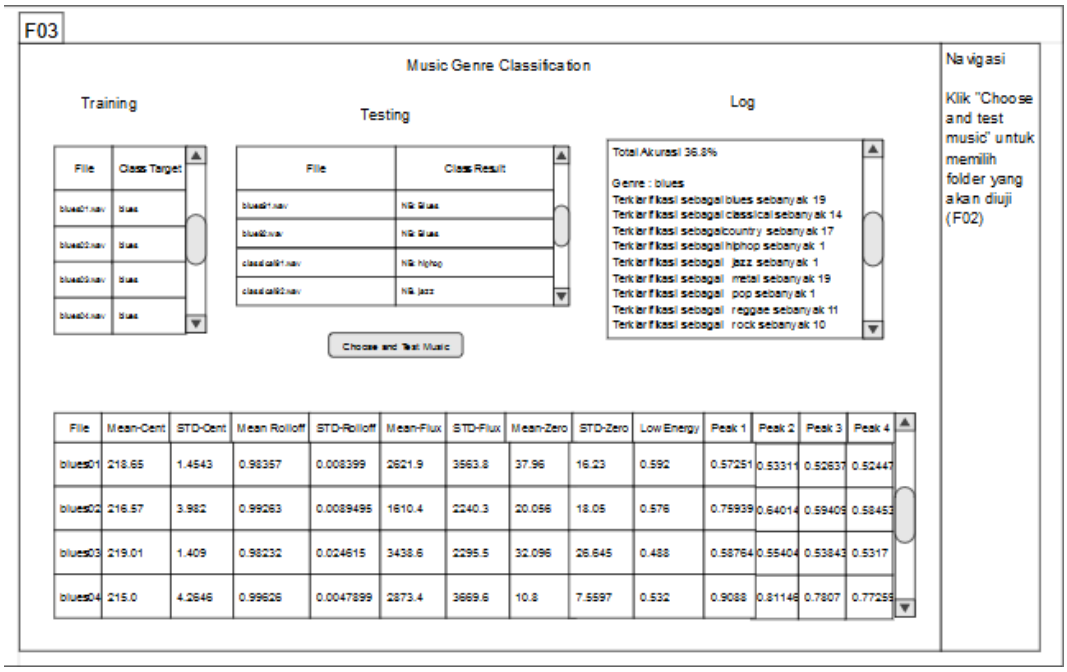
Untuk memilih folder data yang akan diujikan, tekan *choose and test music* dan pilih folder data *testing* yang ingin diklasifikasi genre musiknya. Seperti yang ditunjukkan pada gambar 3.33.



Gambar 3. 33 F02 Pilih Folder Data *Testing*

3. F03 – *Form* hasil klasifikasi

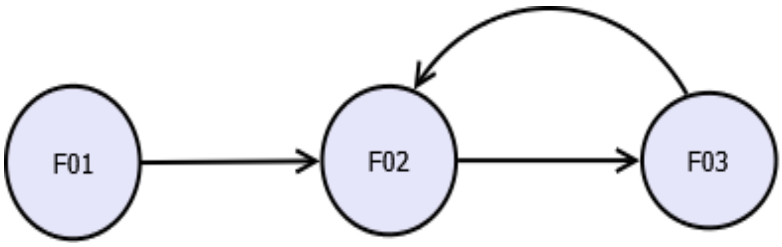
Setelah dilih folder data *testingnya*, data musik tersebut akan diklasifikasikan dan menghasilkan data sesuai dengan kelas hasil klasifikasinya seperti yang ditunjukkan pada gambar 3.34.



Gambar 3. 34 F03 Hasil Klasifikasi

3.6.2 Jaringan Semantik

Jaringan semantik menggambarkan keterhubungan dari navigasi menu antara satu halaman dengan halaman yang lain. Keterhubungan dari navigasi pada setiap menu dari sistem ini dapat dilihat pada gambar 3.35.



Gambar 3. 35 Jaringan Semantik