



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Dalam melakukan penelitian ini, penulis menggunakan langkah kerja sebagai berikut.

1. Melakukan studi pustaka untuk implementasi pada metode mel-frequency cepstral coefficient yang digunakan untuk melakukan ekstraksi ciri suara. Pada penelitian ini, proses MFCC dilakukan di Matlab, sehingga *code* untuk melakukan proses ekstraksi ciri merupakan dari sumber yang sudah pernah menggunakannya dan penulis melakukan beberapa revisi guna menyesuaikan kebutuhan dari code pada penelitian ini. Melakukan studi pustaka untuk implementasi algoritma *K-Nearest Neighbor* sebagai pengolah dari hasil ekstraksi ciri suara. Mencari tahu bagaimana cara kerja dari algoritma *K-Nearest Neighbor* dari beberapa penelitian yang juga menggunakan algoritma ini.
2. Mengambil data sampel berupa hasil dari rekaman jenis suara vokal manusia (*sopran, alto, tenor, dan bass*) pada pria dan wanita. Jumlah dari masing-masing jenis suara adalah sebanyak tiga orang pada masing-masing jenis suara. Proses perekaman data dilakukan di sebuah ruang laboratorium audio di mana suara direkam menggunakan *microphone* khusus sehingga suara yang dihasilkan cukup baik. Proses perekaman dilakukan secara terpisah di mana satu orang anggota paduan suara akan menyanyikan tangga

nada (do, re, mi, fa, sol, la, si, do') tanpa diiringi alat musik. Pengambilan data dilakukan sebanyak tiga kali pengucapan tangga nada untuk masing – masing jenis suara.

3. Data rekaman suara yang sudah diambil kemudian disimpan dalam format *.wav agar dapat diproses. Penyimpanan data suara kedalam format *audio .wav* di karenakan format suara tersebut merupakan *uncompressed sound file*. Data suara asli di mana di dalamnya terdapat rekaman tiga kali pengucapan tangga nada diubah menjadi data-data kecil di mana data asli berupa satu *file* suara dengan tiga kali pengucapan tangga nada, diubah menjadi dua puluh empat (24) *file* suara (8 kata x 3 kali pengucapan).
4. Melakukan ekstraksi ciri suara menggunakan *Mel-Frequency Cepstral Coefficient* (MFCC) di mana hasilnya berupa matriks yang disimpan pada variabel “*melCeps*” yang berbentuk array dua dimensi dengan tipe data *double*. Hasil masing-masing “*melCeps*” tersebut yang kemudian akan digunakan untuk melakukan pelatihan pada *K-Nearest Neighbor* (K-NN).
5. Mendesign sistem berupa *prototype* yang akan dibutuhkan untuk melakukan dekteksi jenis suara vokal. Membuat design awal untuk aplikasi pendeteksi suara vokal.
6. Melakukan proses pelatihan dan menghitung nilai rata dari “*MelCeps*” yang dihasilkan pada data sampel dan diberi label sesuai dengan jenis nada yang digunakan, yaitu : do, re, mi, fa, sol, la, si. Hasil dari rata – rata tersebut disimpan dalam database yang kemudian akan digunakan untuk mencari nilai atau jarak terdekatnya.

7. Melakukan proses pemrograman di mana penulisan kode-kode akan menggunakan bahasa pemrograman PHP untuk algoritma K-NN dan MFCC.
8. Melakukan pengujian terhadap aplikasi yang dibuat.
Setelah selesai melakukan proses pemrograman pada aplikasi, hasilnya akan dilakukan uji coba untuk mengecek apakah hasil yang dihasilkan sesuai dengan data yang ada dan apakah aplikasi berjalan dengan baik.
9. Melihat hasil dari algoritma K-NN seberapa tepat hasil deteksi atau klasifikasi yang dihasilkan.

3.2 Perancangan Aplikasi

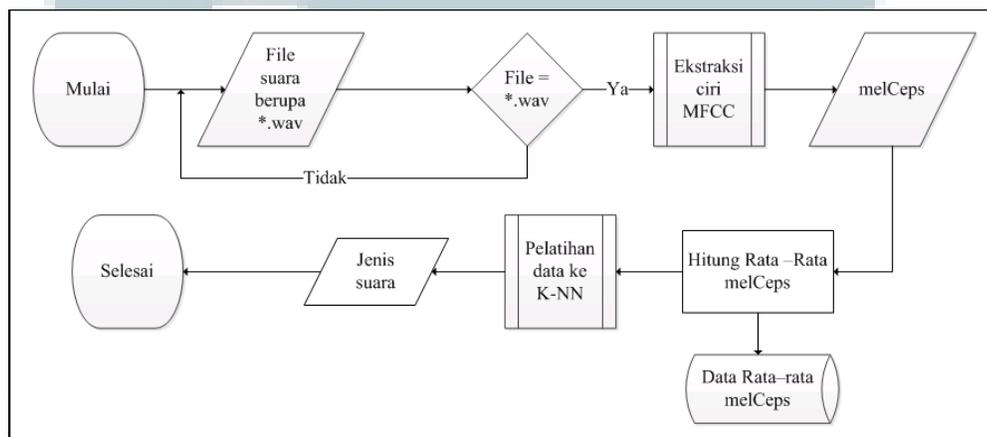
Aplikasi ini dirancang untuk dapat mengambil data berupa suara vokal manusia di mana kemudian akan diproses sehingga menghasilkan informasi berupa jenis suara vokal apa yang dimiliki oleh orang tersebut. Suara yang diterima oleh aplikasi ini merupakan suatu pengucapan tangga nada.

3.2.1 Flowchart

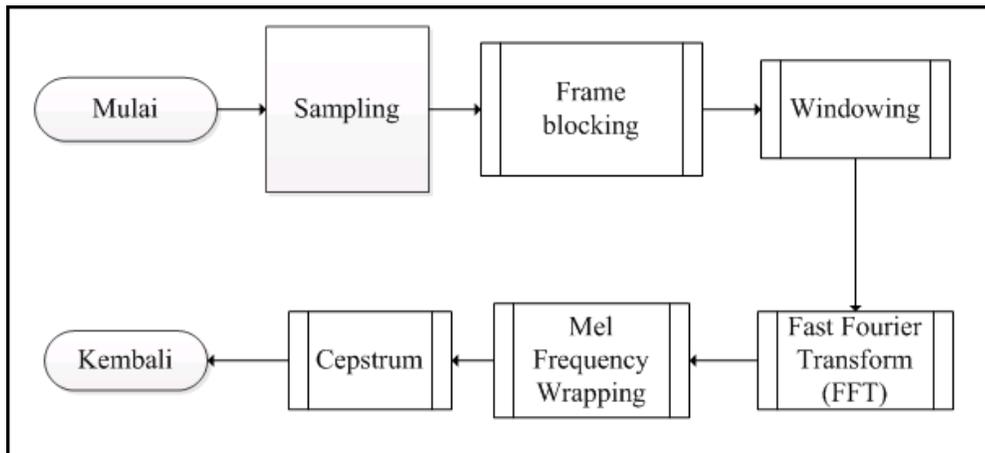
Flowchart diagram pada penelitian ini, dimulai dari memberikan inputan berupa *file* suara berformat *.wav yang telah dikumpulkan. *File* data tersebut akan dicek apakah ada kesalahan dalam perekaman suaranya sehingga hasil datanya tidak sesuai. Jika data yang diinput sudah sesuai maka data tersebut akan mulai diolah menggunakan proses ekstraksi ciri suara menggunakan MFCC.

Proses MFCC ini dilakukan menggunakan PHP dengan bantuan MATLAB untuk membaca file wav, di mana data dari *file* wav tersebut diubah dari *file* yang berupa sinyal analog menjadi sinyal digital. Hasil akhir dari proses MFCC ini berupa suatu kelompok data angka bernama melCeps.

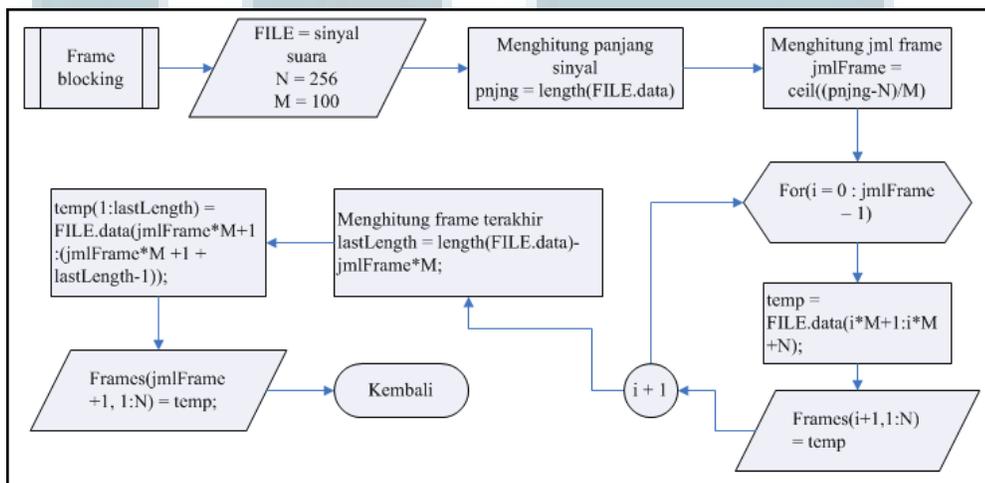
Kelompok data tersebut disimpan ke dalam variabel berupa array dua dimensi (matriks). Variabel tersebut yang akan digunakan untuk melakukan perhitungan rata – rata dan proses penghitungan jarak terdekat pada K-NN. Hasil dari perhitungan rata – rata yang akan disimpan kedalam *database* kemudian hasil tersebut digunakan untuk mendeteksi atau mengklasifikasikan jenis suara vokal manusia.



Gambar 3.1 Flowchart untuk Proses Mendeteksi Jenis Suara

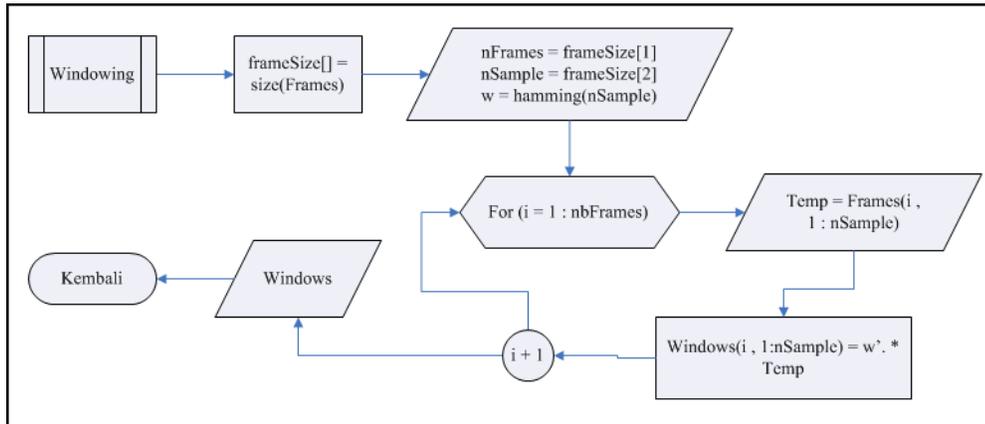


Gambar 3.2 Flowchart Ekstraksi ciri MFCC



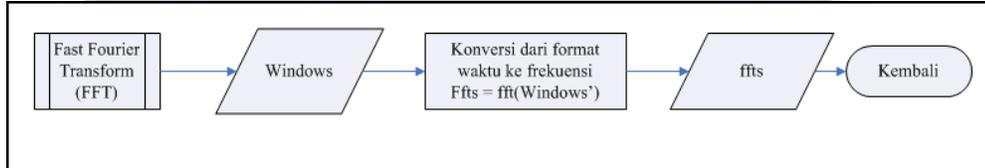
Gambar 3.3 Flowchart Frame Blocking

Gambar 3.3 di atas menampilkan gambar dari proses *Frame Blocking* yang terjadi dalam sistem. Inputan yang diberikan berupa sinyal suara, besar N dan M. setelah itu melakukan proses perhitungan panjang sinyal agar dapat mengetahui berapa jumlah frame yang akan terbentuk. Selanjutnya adalah melakukan *looping* untuk membagi sinyal suara ke dalam *frame – frame*.



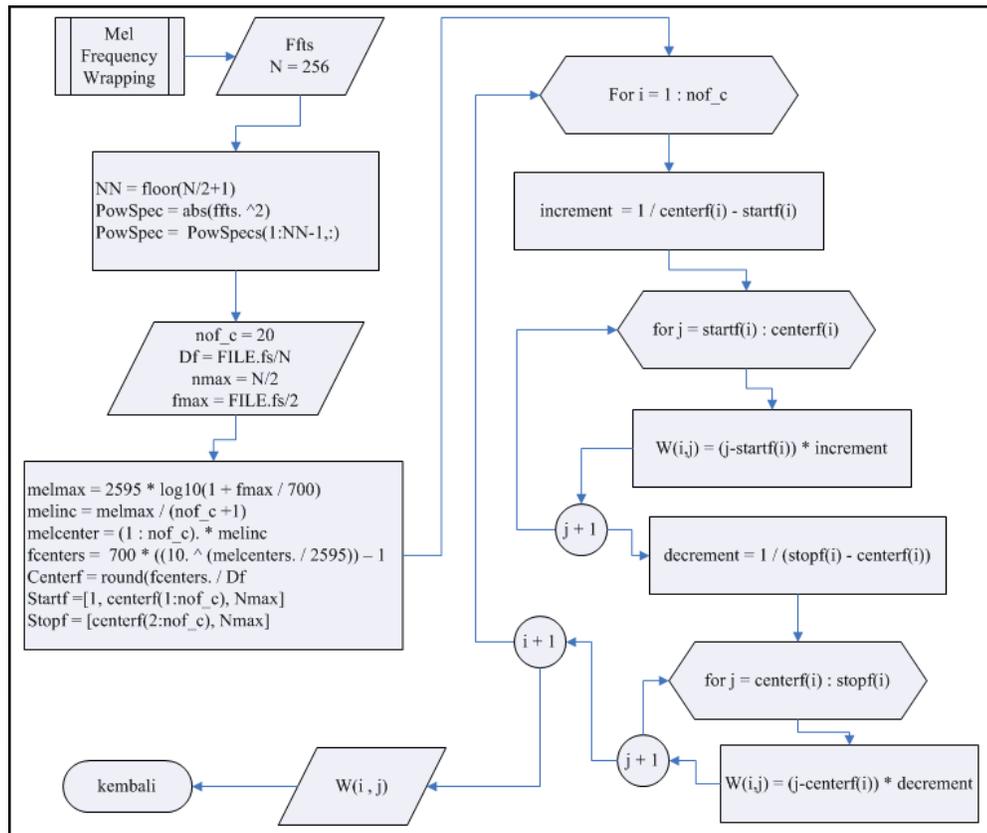
Gambar 3.4 *Flowchart* Windowing

Gambar di atas menampilkan proses dari *windowing*, proses ini melakukan proses pengerucutan pada akhir *frame* yang terbentuk pada proses *frame blocking* sehingga meminimalisir terjadinya diskontinuitas sinyal pada awal dan akhir *frame*.



Gambar 3.5 *Flowchart* Fast Fourier Transform (FFT)

Gambar 3.5 menampilkan proses dari FFT di mana data yang sudah dilakukan windowing diubah formatnya yang sebelumnya berada dalam format waktu dikonversikan ke dalam format frekuensi.

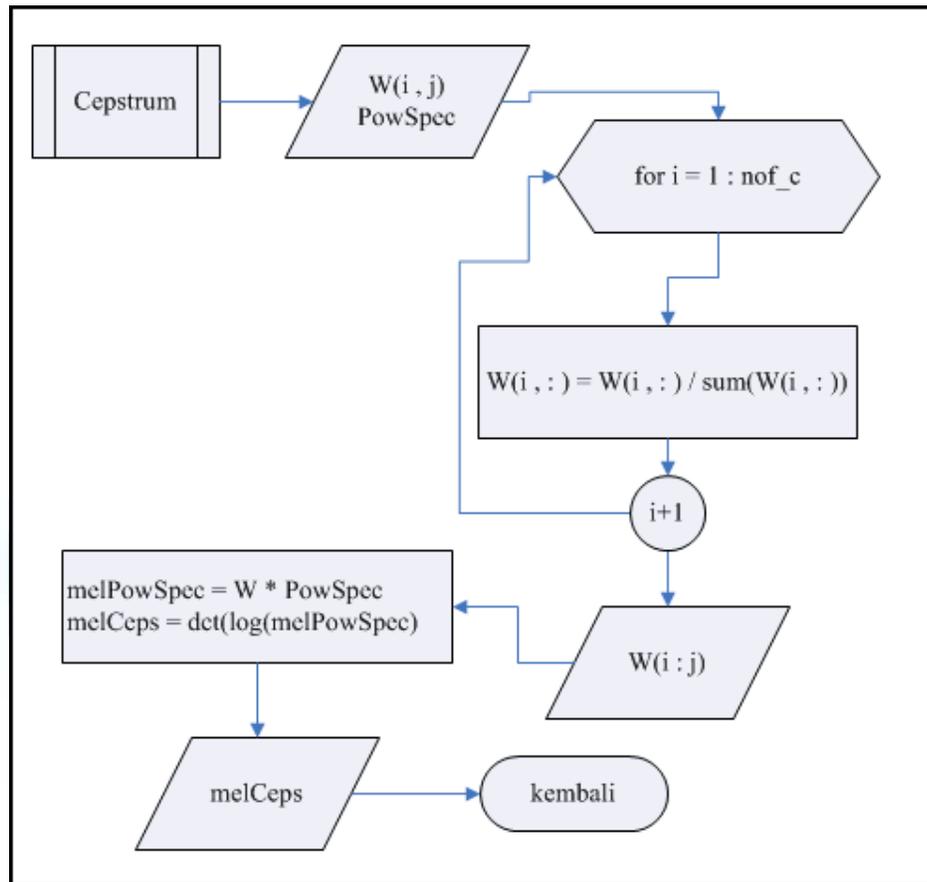


Gambar 3.6 Flowchart MelFrequency Wrapping

Pada gambar di atas, hasil dari proses FFT selanjutnya melakukan proses mengukur dan mengubah nada-nada yang memiliki frekuensi sesungguhnya diubah menjadi sebetuk pola yang diukur dalam skala

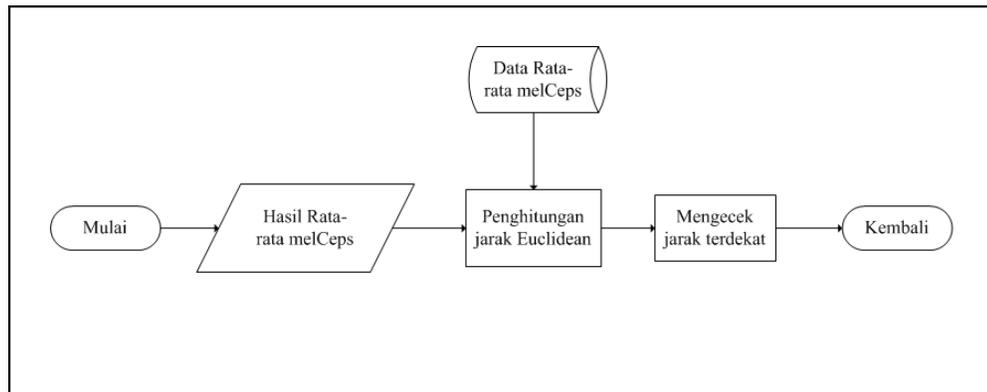
“mel”.





Gambar 3.7 Flowchart Cepstrum

Proses terakhir, yaitu *cepstrum* merupakan proses untuk mengubah skala “*mel*” yang dihasilkan pada proses sebelumnya menjadi *cepstrum* menggunakan perhitungan *Discrete Cosine Transform* (DCT). Hasilnya berupa file data yang kemudian disimpan dalam suatu variabel. Hasil ini yang dinamakan MFCC.



Gambar 3.8 *Flowchart* pelatihan data ke K-NN

Diagram 3.8 menunjukkan proses dari K-NN di mana data melCeps yang telah didapatkan dari proses ekstraksi ciri jenis suara menggunakan metode MFCC dilakukan. Data melCeps tersebut terlebih dahulu dihitung jumlah rata-ratanya. Setelah rata-rata dari data baru tersebut didapatkan, data latih diambil dari database sesuai dengan kebutuhan (contoh : hanya *alto* dan *sopran* saja atau *tenor* dan *bass* saja).

Rata-rata dari data yang baru dan rata-rata pada data latih tersebut kemudian diproses menggunakan rumus *Euclidean distance* untuk mencari tahu jaraknya. Setelah hasil dari penghitungan jarak didapatkan, maka dapat ditentukan hasil dari perhitungan mana yang memiliki jarak terdekat.

3.2.2 Tabel Data

Penelitian ini data latih dan data penting yang diperlukan disimpan dalam sebuah *database* yang di dalamnya terdiri dari lima belas (15) *tables* yaitu, *jenis_suar*, *melcepsalto1_detail*, *melcepsalto2_detail*, *melcepsalto3_detail*, *melcepsbass1_detail*, *melcepsbass2_detail*, *melcepsbass3_detail*, *melcepsopran1_detail*, *melcepsopran2_detail*, *melcepsopran3_detail*,

melcepsstenor1_detail, melcepsstenor2_detail, melcepsstenor3_detail, melceps_header, dan rata_melceps_suara. Berikut ini merupakan struktur dari masing – masing *table*.

Nama tabel : jenis_suara

Fungsi : menyimpan data-data jenis suara

Tabel 3.1 jenis_suara

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id_jenisSuara	int	11	Primary Key
2	jenisSuara	varchar	10	

Nama tabel : melcepsalto1_detail

Fungsi : menyimpan data melCeps data latihan yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara alto, di mana merupakan pengambilan data untuk jenis suara alto ke-1.

Tabel 3.2 melcepsalto1_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsalto2_detail

Fungsi : menyimpan data melCeps data latihan yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang

disimpan di sini adalah data melceps untuk jenis suara alto, di mana merupakan pengambilan data untuk jenis suara alto ke-2.

Tabel 3.3 melcepsalto2_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsalto3_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara alto, di mana merupakan pengambilan data untuk jenis suara alto ke-3.

Tabel 3.4 melcepsalto3_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsbass1_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara bass, di mana merupakan pengambilan data untuk jenis suara bass ke-1.

Tabel 3.5 melcepsbass1_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsbass2_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara bass, di mana merupakan pengambilan data untuk jenis suara bass ke-2.

Tabel 3.6 melcepsbass2_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsbass3_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara bass, di mana merupakan pengambilan data untuk jenis suara bass ke-3.

Tabel 3.7 melcepsbass3_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsbass3_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara sopran, di mana merupakan pengambilan data untuk jenis suara sopran ke-1.

Tabel 3.8 melcepsbass2_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepsbass2_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara sopran, di mana merupakan pengambilan data untuk jenis suara sopran ke-2.

Tabel 3.9 melcepsbass1_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepssopran3_detail

Fungsi : menyimpan data melCeps data latihan yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara sopran, di mana merupakan pengambilan data untuk jenis suara sopran ke-3.

Tabel 3.10 melcepssopran3_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepstenor1_detail

Fungsi : menyimpan data melCeps data latihan yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara tenor, di mana merupakan pengambilan data untuk jenis suara tenor ke-1.

Tabel 3.11 melcepstenor1_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepstenor2_detail

Fungsi : menyimpan data melCeps data latihan yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang

disimpan di sini adalah data melceps untuk jenis suara tenor, di mana merupakan pengambilan data untuk jenis suara tenor ke-2.

Tabel 3.12 melcepstenor2_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melcepstenor3_detail

Fungsi : menyimpan data melCeps data latih yang diperoleh dari hasil ekstraksi ciri suara menggunakan MFCC. Data melCeps yang disimpan di sini adalah data melceps untuk jenis suara tenor, di mana merupakan pengambilan data untuk jenis suara tenor ke-3.

Tabel 3.13 melcepstenor3_detail

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	melCeps	double		

Nama tabel : melceps_header

Fungsi : menyimpan data header dari *field-field* pada melceps_detail. Data yang disimpan berupa jenis nada yang direkam. Tabel ini memiliki id sebagai *primary key* untuk dapat menghubungkan data yang ada pada tabel header dan data yang ada pada tabel detail.

Tabel 3.14 melceps_header

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	Id	int	11	Primary Key
2	nadaSuara	varchar	5	

Nama tabel : rata_melceps_suara

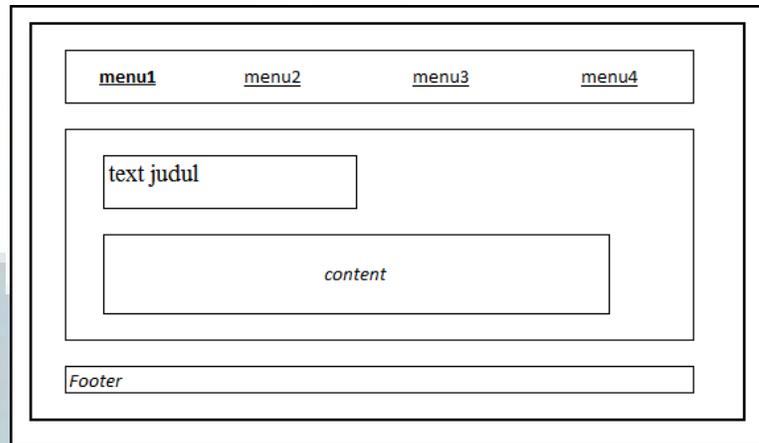
Fungsi : menyimpan semua data hasil dari perhitungan rata-rata melCeps pada masing – masing jenis nada yang ada. Tabel ini berhubungan dengan tabel melceps_header dan tabel jenis_suara atau dapat disebut juga merupakan tabel detail dari dua tabel header tersebut.

Tabel 3.15 rata_melceps_suara

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id_rata_melceps	int	11	Primary Key
2	id_nada_suara	int	11	
3	id_jenisSuara	int	11	
4	rata_rata_melceps	double		

UMMN

3.2.3 Rancangan *User Interface*

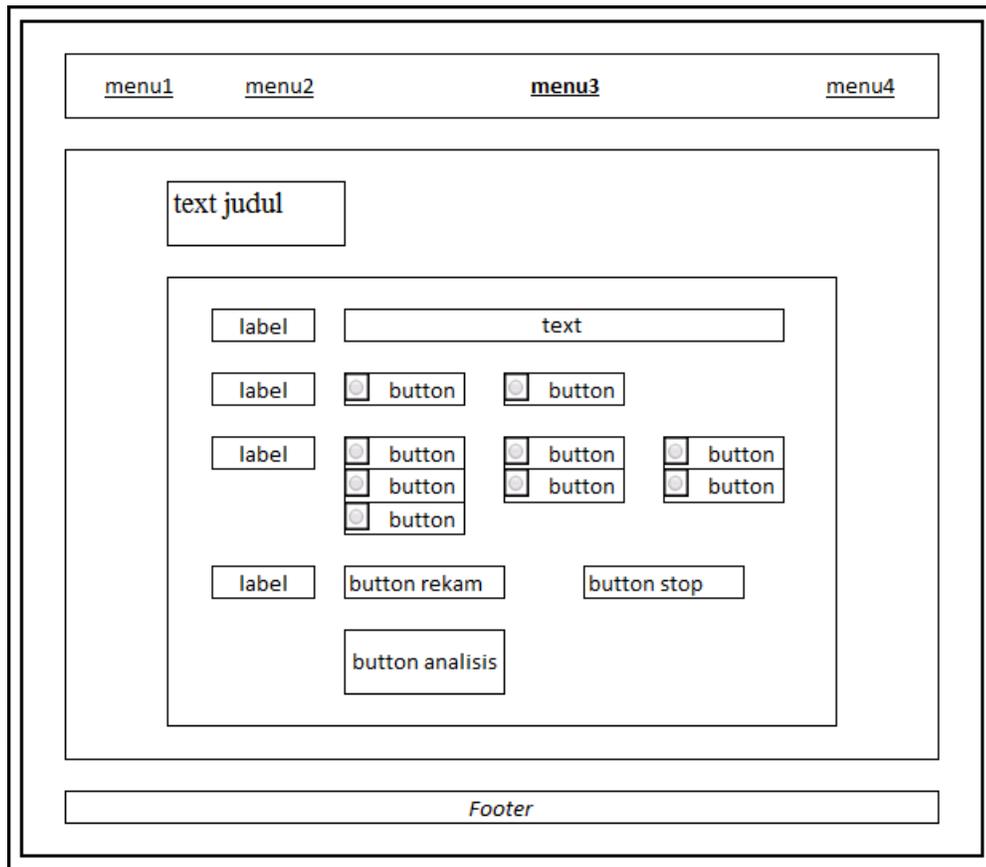


Gambar 3.9 Rancangan untuk Tampilan Menu1, Menu2, Menu4

Rancangan *interface* yang akan dibangun pada penelitian ini seperti pada gambar 3.9 di mana terdapat beberapa menu yang akan mengarahkan pengguna pada halaman-halaman web lainnya. Juga terdapat kata pengantar yang menjelaskan secara singkat mengenai web tersebut.

Selain itu, gambar 3.9 juga merupakan rancangan untuk halaman bantuan yang akan dibangun. Bedanya adalah pada bagian *text* judul dan *context*. Pada *context* berisi tentang bagaimana cara menggunakan aplikasi untuk mendeteksi jenis suara vokal ini. Cara – cara penggunaannya akan dijelaskan secara singkat dan mudah dimengerti.

Gambar 3.9 juga merupakan rancangan untuk menampilkan beberapa informasi mengenai jenis – jenis suara yang ada pada menu4. Informasi ini diberikan sebagai pelengkap dari aplikasi pendeteksi jenis suara ini sehingga *user* dapat mengetahui secara singkat mengenai jenis suara yang dimilikinya.



Gambar 3.10 Halaman Rekam Suara

Pada gambar 3.10 terlihat rancangan *interface* untuk melakukan pengambilan suara (merekam) manusia yang akan dideteksi termasuk jenis suara vokal apakah yang dimiliki. Pengguna diharapkan mengisi nama dan jenis kelamin kemudian memilih nada suara apa yang akan dinyanyikan.

Proses pengambilan suara / melakukan rekaman suara akan berjalan ketika *button* yang berfungsi untuk melakukan rekaman ditekan. Untuk menghentikan proses perekaman, *button stop* harus diklik. Setelah *button stop* ditekan, maka file suara yang sudah direkam tadi akan disimpan

kedalam suatu folder yang nantinya akan digunakan dalam proses pendeteksi. Suara yang direkam tersebut langsung tersimpan dalam format wav.

Proses analisis jenis suara akan dimulai ketika *user* menekan *button* analisis. Pada proses ini file data suara yang tadi sudah disimpan diproses dimulai dengan dilakukan ekstraksi ciri suara lalu kemudian melakukan proses deteksi menggunakan algoritma K-NN.

The logo of Universitas Muhammadiyah Negeri (UMMN) is displayed. It consists of a circular emblem with a stylized face or mask inside, and the letters 'UMMN' written in a bold, blue, sans-serif font below it.