



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

METODE DAN PERANCANGAN SISTEM

3.1 Metode Penelitian

Metode-metode yang digunakan dalam penelitian ini antara lain sebagai berikut.

1. Studi literatur

Studi literatur dilakukan dengan membaca buku, artikel, dan hasil penelitian lain yang berhubungan dengan penelitian ini, terutama mengenai pengolahan citra digital dan algoritma *K-Nearest Neighbors* dan *Manhattan Distance*.

2. Pengumpulan data

Pengumpulan data dilakukan dengan mengambil data kematian ikan tiap jam dengan mengambil gambar ikan menggunakan kamera digital.

3. Perancangan aplikasi

Perancangan aplikasi dilakukan dengan membuat *Data Flow Diagram* (DFD), *flowchart*, struktur tabel dan rancangan antarmuka dari aplikasi.

4. Pemrograman

Pemrograman dilakukan dengan menuliskan kode-kode dalam bahasa pemrograman untuk membangun aplikasi ini.

5. *Testing*

Testing dilakukan dengan melakukan percobaan terhadap sistem untuk mencari *error* atau kekekuran lain yang ada pada aplikasi.

6. Pengujian

Pengujian terhadap aplikasi dilakukan dengan menjalankan aplikasi yang sudah dibuat, lalu membandingkan hasilnya data waktu kematian ikan yang sudah dikumpulkan untuk mengetahui akurasi dari aplikasi.

7. Analisis

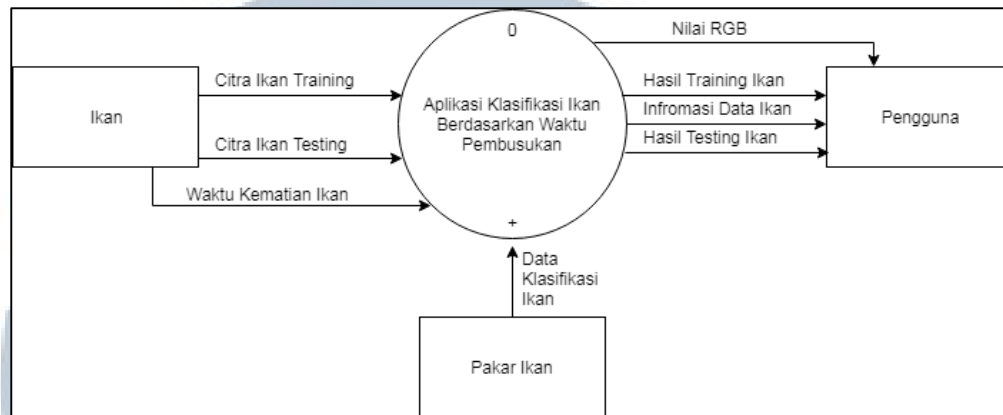
Analisis dilakukan dengan menuliskan hasil pengujian terhadap aplikasi ke dalam laporan.

3.2 Perancangan Aplikasi

Perancangan terhadap aplikasi yang telah dibuat digambarkan dalam beberapa diagram sebagai berikut. Beserta dengan pengambilan data berupa gambar atau foto ikan nilai berjumlah 30 gambar sebagai sampel data. Menurut Roscoe (1975), disarankan untuk sampel minimal dalam sebuah penelitian adalah 30 data secara umum, dimana 10 sampel data terlalu sedikit untuk sebuah penelitian. Pengambilan foto ikan diambil di ruangan yang sama dengan suhu ruangan 29-30 derajat Celcius.

3.2.1 Data Flow Diagram

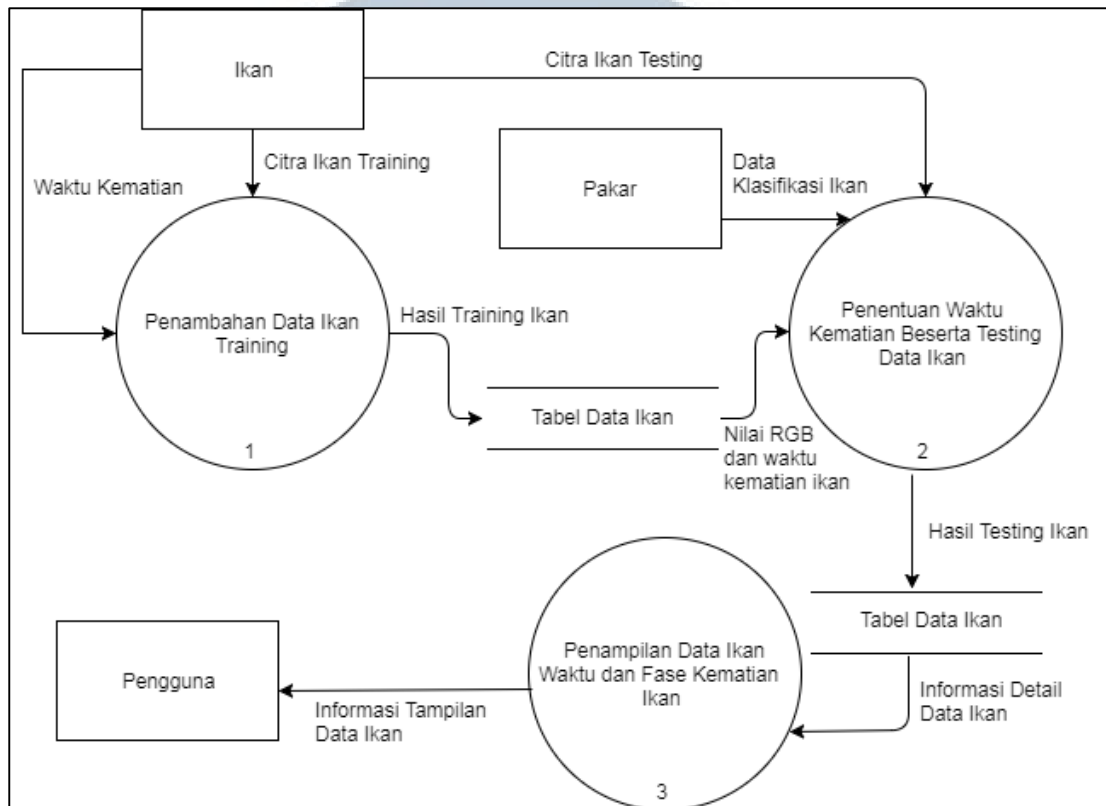
Pada perancangan terhadap *Data Flow Diagram* (DFD) dimulai dari perancangan *Data Flow Diagram Level 0*. Berikut merupakan perancangan DFD *Level 0* yang ditampilkan melalui Gambar 3.1.



Gambar 3.1 Data Flow Diagram Level 0

Pada DFD *Level 0* digambarkan aliran data pada aplikasi klasifikasi ikan berdasarkan waktu pembedaan secara keseluruhan. Terdapat sebuah entitas bernama ikan dimana memberikan masukan untuk aplikasi dan memberikan keluaran yang diterima oleh pengguna. Keluaran yang diberikan oleh aplikasi merupakan informasi hasil *training* dan *testing* data sehingga dapat ditentukan sebuah informasi berupa klasifikasi ikan berdasarkan waktu pembedaannya.

Selanjut pada DFD *Level 1* akan dijelaskan lebih rinci cara kerja aplikasi ini untuk menentukan klasifikasi ikan. Berikut merupakan tampilan DFD *Level 1* yang dapat dilihat melalui Gambar 3.2.



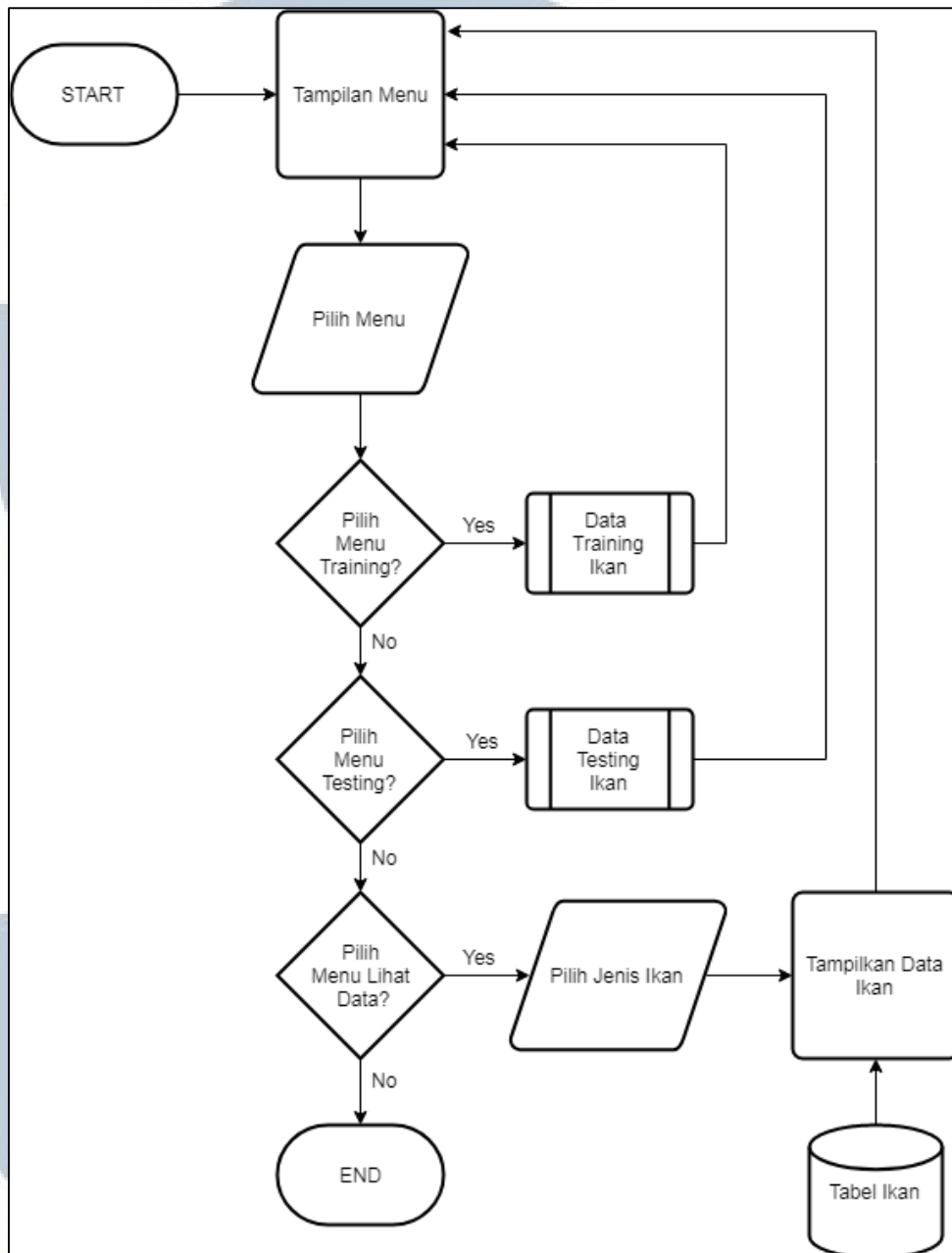
Gambar 3.2 Data Flow Diagram Level 1

Pada DFD *Level 1*, aplikasi klasifikasi ikan dapat dibagi menjadi tiga subproses yang berhubungan dengan sebuah tabel bernama “Tabel Data Ikan”. Subproses yang pertama adalah “Penambahan Data Ikan *Training*”, dimana data *citra ikan training* dan *waktu* dialirkan dari entitas ikan untuk melakukan proses *training*. Data tersebut akan diproses sehingga data akan berubah menjadil *hasil ikan training* yang akan disimpan pada “Tabel Data Ikan”. Pada subproses “Penentuan Waktu Kematian Beserta *Testing* Data Ikan” akan menerima data *citra ikan testing* dan akan diproses beserta hasil *training* berupa waktu dan nilai RGB. Dalam subproses ini *testing* data akan dilakukan dengan menggunakan algoritma *K-Nearest Neighbors* dengan perhitungan distance

menggunakan Manhattan Distance, untuk mencari waktu kematian ikan dari jarak terdekat dari data *training* yang disimpan. Hasil proses tersebut akan menjadi data *testing* yang akan disimpan. Subproses terakhir adalah “Penampilan Data Ikan Waktu dan Fase Kematian”, dimana subproses ini membutuhkan aliran data yang disimpan dari tabel. Data tersebut akan diproses dan ditampilkan sehingga dapat dilihat oleh pengguna.

3.2.2 Flowchart

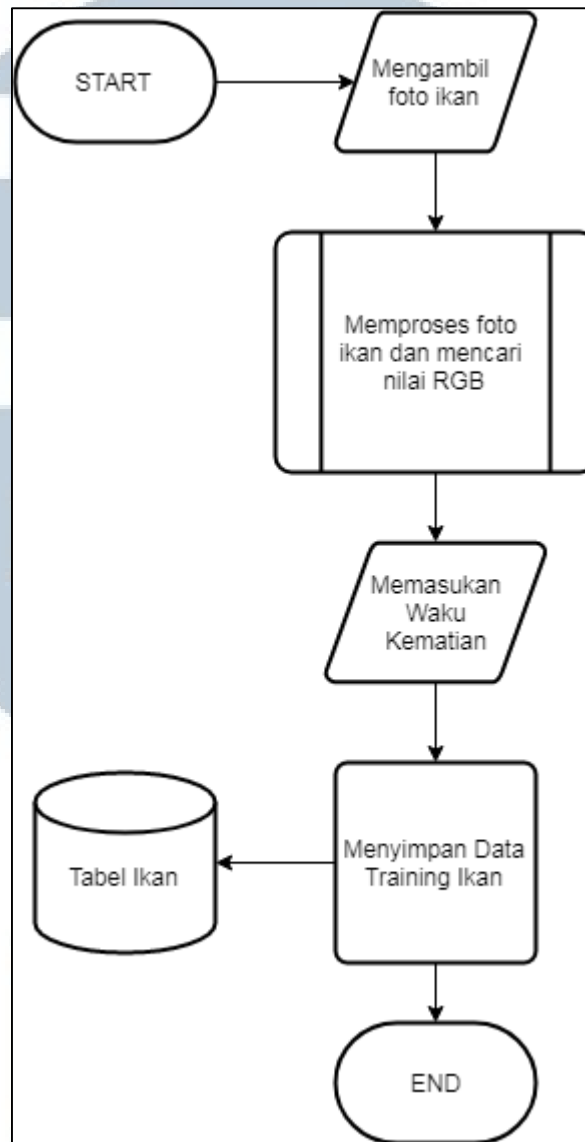
Pada perancangan *flowchart* secara keseluruhan akan dijelaskan aliran proses atau tahap-tahap yang terjadi dalam penelitian aplikasi ini. Berikut merupakan *flowchart* secara keseluruhan pada aplikasi yang dibuat pada penelitian ini. *Flowchart* dapat dilihat pada Gambar 3.3. Pada *flowchart* aplikasi ini terdapat dua proses, yaitu data ikan *training* dan data ikan *testing*. Proses tersebut akan dijelaskan lebih lanjut pada *flowchart* tersendiri. Pada *flowchart* utama aplikasi dimana ketika aplikasi dimulai akan muncul display halaman utama beserta dengan menu pilihan. Pilihan pertama akan masuk ke pilihan proses data ikan *training* atau tidak. Jika iya, akan masuk pada pilihan menu data *training* yang akan dijelaskan lebih rinci pada *flowchart* berikutnya. Setelah proses data ikan *training* selesai, akan kembali ke pilih menu pada tampilan awal. Jika tidak, akan berlangsung ke pilihan menu selanjutnya, yaitu data ikan *testing*. Pilihan menu selanjutnya akan masuk ke proses data ikan *testing* atau tidak.



Gambar 3.3 *Flowchart* Aplikasi Keseluruhan

Jika iya, akan masuk pada pilihan menu data *testing* yang akan dijelaskan lebih rinci pada *flowchart* berikutnya. Setelah proses data *testing* selesai, akan kembali ke pilih menu pada tampilan awal. Jika tidak, akan berlangsung ke pilihan menu selanjutnya,

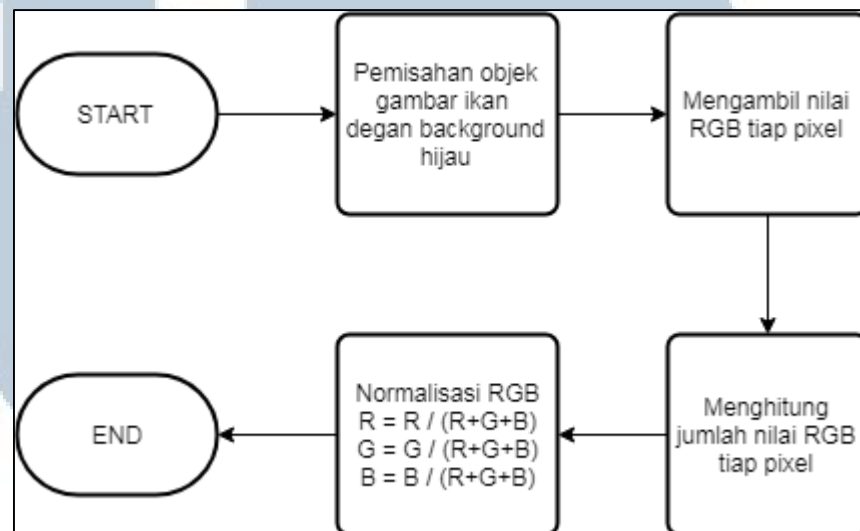
yaitu Lihat Data Ikan. Pilihan menu yang terakhir berfungsi untuk menampilkan data yang sudah disimpan. Kembali pada pilihan selanjutnya, yaitu jika memilih iya, akan masuk ke proses Lihat Data Ikan. Sebelum proses untuk menampilkan data, dibutuhkan untuk memasukan jenis ikan untuk menampilkan data ikan. Setelah itu, proses dapat menampilkan data keseluruhan dari *database* yang sudah disimpan pada proses *training* sebelumnya. Ketika proses selesai akan kembali ke pilih menu pada tampilan awal. Jika tidak, aplikasi selesai bekerja. Untuk penjelasan lebih lanjut seperti pada proses data *training* ikan dan data *testing* ikan akan dijelaskan rinci pada *flowchart-flowchart* berikutnya. Semua pilihan pada proses tampilan menu ini jika sudah berakhir akan kembali lagi pada tampilan menu bagian awal. Sehingga aplikasi akan selesai bekerja setelah tidak ada pilihan proses data *training*, proses data *testing* dan proses tampilkan data ikan. Itulah penjelasan aliran proses yang terjadi pada aplikasi secara keseluruhan. Pada *flowchart* berikutnya, akan dijelaskan aliran data yang terjadi setelah memilih proses data *training* pada pilihan menu utama yang sudah disediakan di aplikasi. Berikut merupakan tampilan *flowchart* data *training* ikan yang ditampilkan pada Gambar 3.4.



Gambar 3.4 *Flowchart Data Training*

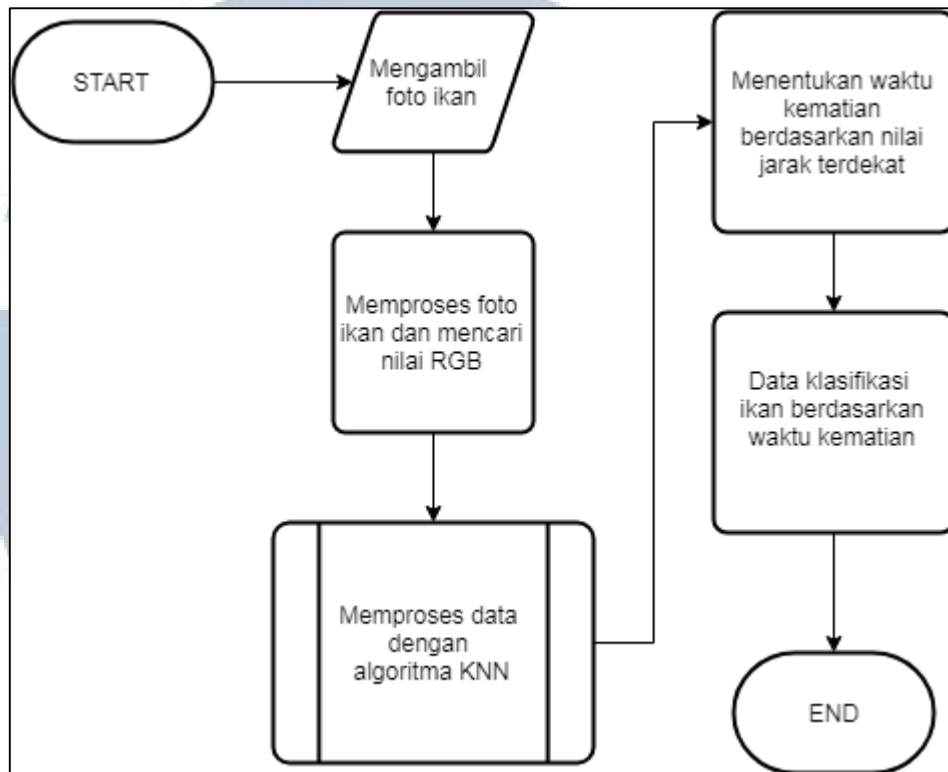
Pada *flowchart* ini, akan dijelaskan proses yang terjadi pada data *training* ikan. Proses aliran data dimulai saat menerima masukan foto atau gambar ikan. Kemudian akan diproses dan diambil nilai RGB dari gambar tersebut yang akan dijelaskan pada *flowchart* selanjutnya. Setelah itu memberikan masukan waktu kematian ikan tersebut. Data yang sudah dikumpulkan dan diproses akan disimpan pada tabel ikan. Demikian

proses data *training* ikan berlangsung. Flowchart selanjutnya, akan dijelaskan cara kerja segmentasi gambar antara objek ikan dengan latar belakang dan normalisasi RGB pada Gambar 3.5.



Gambar 3.5 Flowchart Segmentasi Objek dan Normalisasi RGB

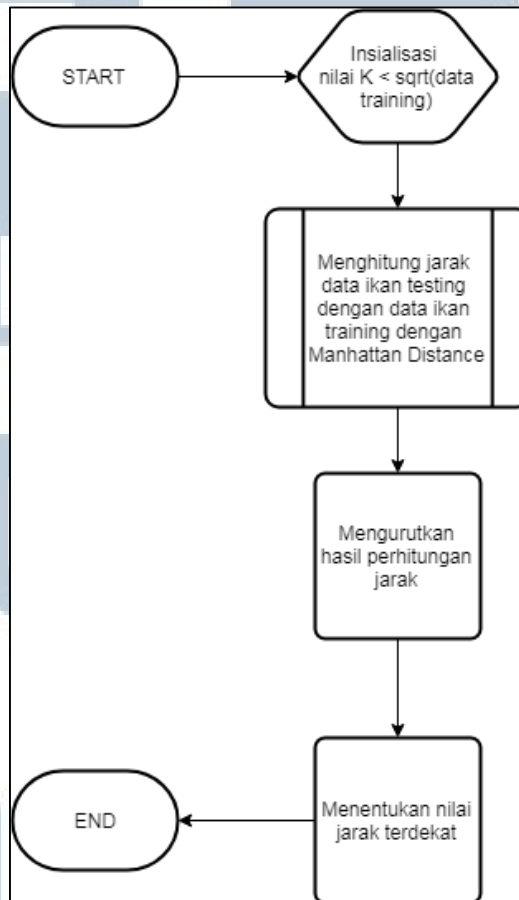
Pada *flowchart* ini, akan dijelaskan proses segmentasi dan normalisasi RGB. Setelah menerima masukan gambar ikan, gambar akan diproses untuk memisahkan ikan sebagai objek dari latar belakang berwarna hijau. Setelah itu akan diambil nilai R, G dan B dari gambar ikan yang sudah diproses. Nilai tersebut akan ditampung dan dijumlahkan untuk mendapat total nilai RGB tersebut. Lalu akan dilakukan perhitungan normalisasi untuk menyelesaikan hasil segmentasi dengan warna. Setelah itu hasil perhitungan normalisasi sudah selesai dan dapat melanjutkan ke proses selanjutnya. Selanjutnya merupakan penjelasan proses *testing* ikan yang digambarkan pada Gambar 3.6.



Gambar 3.6 Flowchart Data Testing

Pada *flowchart* ini, akan dijelaskan proses yang terjadi pada data *testing* ikan. Proses aliran data dimulai saat menerima masukan foto atau gambar ikan. Kemudian akan diproses dan diambil nilai RGB dari gambar tersebut seperti pada penjelasan *flowchart* sebelumnya pada proses data *training*. Setelah itu data akan diproses menggunakan algoritma KNN untuk mencari jarak terdekat yang akan dijelaskan lebih lanjut di *flowchart* berikutnya. Perhitungan jarak akan dihitung menggunakan perhitungan jarak *Manhattan Distance*. Hasil jarak terdekat akan ditentukan pada proses berikutnya dan mengambil waktu kematian dari data *training* sesuai dengan cara kerja algoritma KNN. Waktu yang sudah didapat akan diklasifikasikan untuk menentukan fase kematian ikan tersebut. Berikut merupakan *flowchart* dari cara kerja

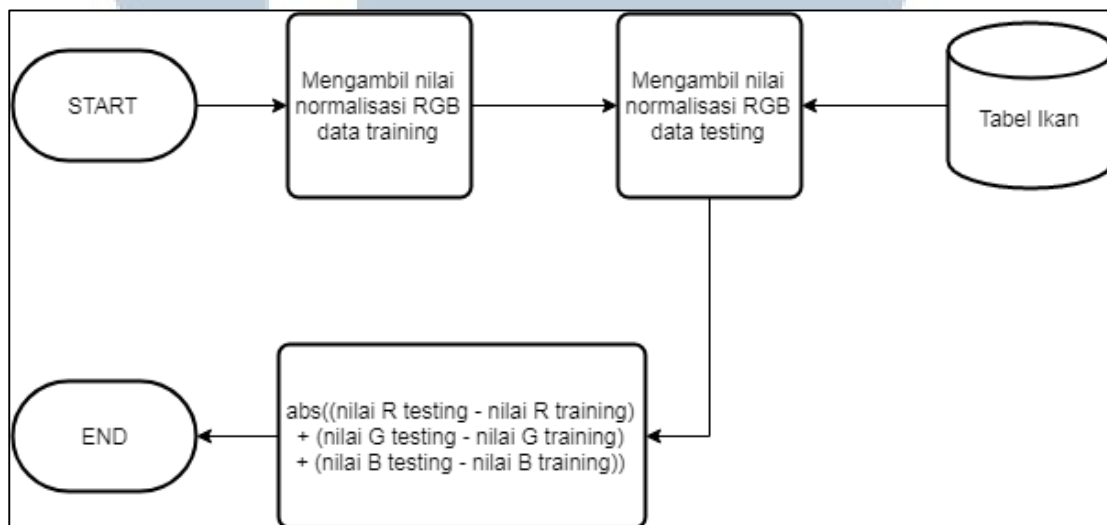
algoritma KNN untuk memproses data ikan yang sudah didapat yang dapat dilihat pada Gambar 3.7 berikut.



Gambar 3.7 Flowchart Algoritma KNN

Pada *flowchart* ini, akan dijelaskan cara kerja algoritma KNN pada proses data *testing* sebelumnya. Aplikasi akan melakukan inisialisasi K untuk menentukan jumlah jarak terdekat dari perhitungan *distance* yang akan dilakukan nanti. Nilai K yang berada pada aplikasi ini adalah $K = 1$. Setelah itu akan dilakukan perhitungan jarak antara data *testing* dengan data *training* yang akan dijelaskan lebih lanjut pada *flowchart* selanjutnya. Setelah perhitungan sudah selesai, data jarak akan diurutkan dari

nilai terkecil hingga terbesar. Lalu aplikasi akan menentukan nilai terkecil sebagai jarak terdekat antara data *testing* dengan data *training* hasil perhitungan *distance* sebelumnya. Jarak terdekat tersebut digunakan untuk mencari waktu kematian ikan dari data *training* untuk proses data *testing* yang sudah dijelaskan sebelumnya. Selanjutnya akan dijelaskan langkah-langkah perhitungan *distance* dengan menggunakan perhitungan *Manhattan Distance*. Berikut merupakan *flowchart* dari perhitungan jarak dengan *Manhattan Distance* yang dapat dilihat dari Gambar 3.8.



Gambar 3.8 *Flowchart* Perhitungan Jarak dengan *Manhattan Distance*

Pada *flowchart* ini, akan dijelaskan bagaimana cara perhitungan *distance* untuk algoritma KNN yang digunakan untuk mencari klasifikasi ikan tersebut. Proses aliran data dimulai setelah mendapatkan nilai normalisasi RGB dari masukan gambar dari data *testing* yang dijelaskan sebelumnya. Lalu proses berlanjut dengan mengambil data yang normalisasi RGB data *training* yang sudah disimpan sebelumnya. Setelah itu akan dilakukan perhitungan jarak antara normalisasi RGB data *testing* dengan data *training*

yang hasilnya selisih antara dua nilai tersebut merupakan nilai jarak antara data-data tersebut. Hasil perhitungan tersebut akan diurutkan dan akan dicari jarak terpendek untuk menentukan waktu kematian ikan yang terdapat pada proses data *testing* yang sudah dijelaskan pada *flowchart* sebelumnya.

3.2.3 Struktur Tabel

Berikut ini adalah nama, fungsi, beserta struktur dari tabel yang digunakan dalam pada aplikasi ini.

1. Nama Tabel : tabel_ikan

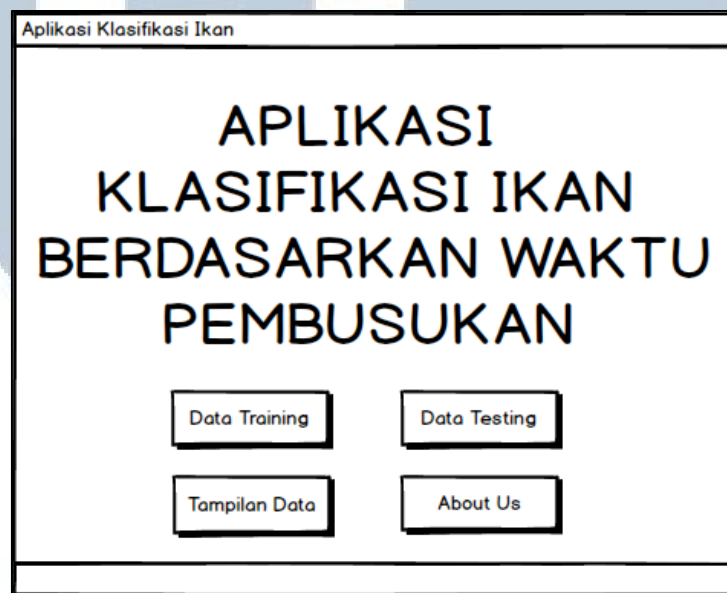
Fungsi : menyimpan data *training* yang sudah diproses.

Tabel 3.1. Tabel Ikan

Nama Kolom	Tipe	Deskripsi
No	int(11)	Nomor data
Nama	varchar(50)	Nama data
Jenis	varchar(10)	Jenis Ikan
R	float	Nominal normalisasi R
G	float	Nominal normalisasi G
B	float	Nominal normalisasi B
Waktu	int(11)	Waktu kematian ikan
Fase	varchar(20)	Fase kematian ikan

3.2.4 Rancangan Antarmuka

Rancangan antarmuka adalah rancangan dasar dari aplikasi yang dikembangkan sehingga memiliki tampilan yang hampir serupa dengan aplikasinya, tetapi dengan wujud sederhana agar lebih mudah untuk dimengerti oleh pengguna. Berikut ini adalah rancangan antarmuka aplikasi yang ditampilkan pada Gambar 3.9.



Gambar 3.9 *Mockup* Tampilan Utama

Pada halaman utama terdapat pilihan-pilihan menu yang akan muncul tampilan tersendiri. Jika pilihan menu data *training* dipilih, akan muncul tampilan seperti pada Gambar 3.10.

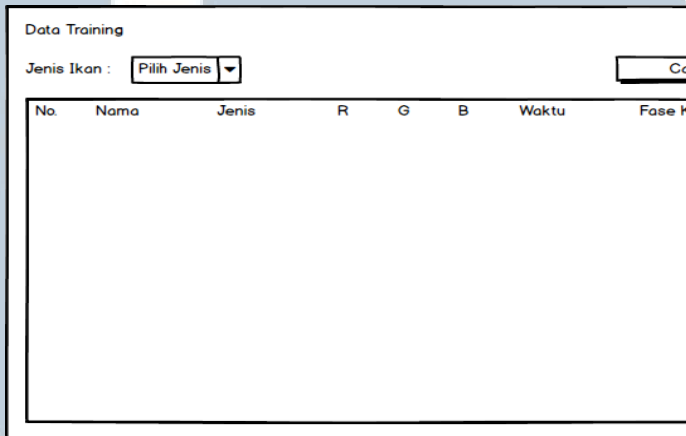
Gambar 3.10 *Mockup Tampilan Form Training Ikan*

Pada form data *training* terdapat bagian\ untuk memproses data yang sudah diterima. Sedangkan tampilan data *testing* dapat dilihat pada Gambar 3.11 berikut ini.

Gambar 3.11 *Mockup Tampilan Form Testing Ikan*

Pada form data *testing* juga terdapat bagian-bagian yang harus diisi dan tombol-tombol untuk memproses data *testing*. Perbedaannya pada form ini akan ditampilkan

g. Berikut merupakan gambar form tampilan data yang digambarkan pada Gambar 3.12.



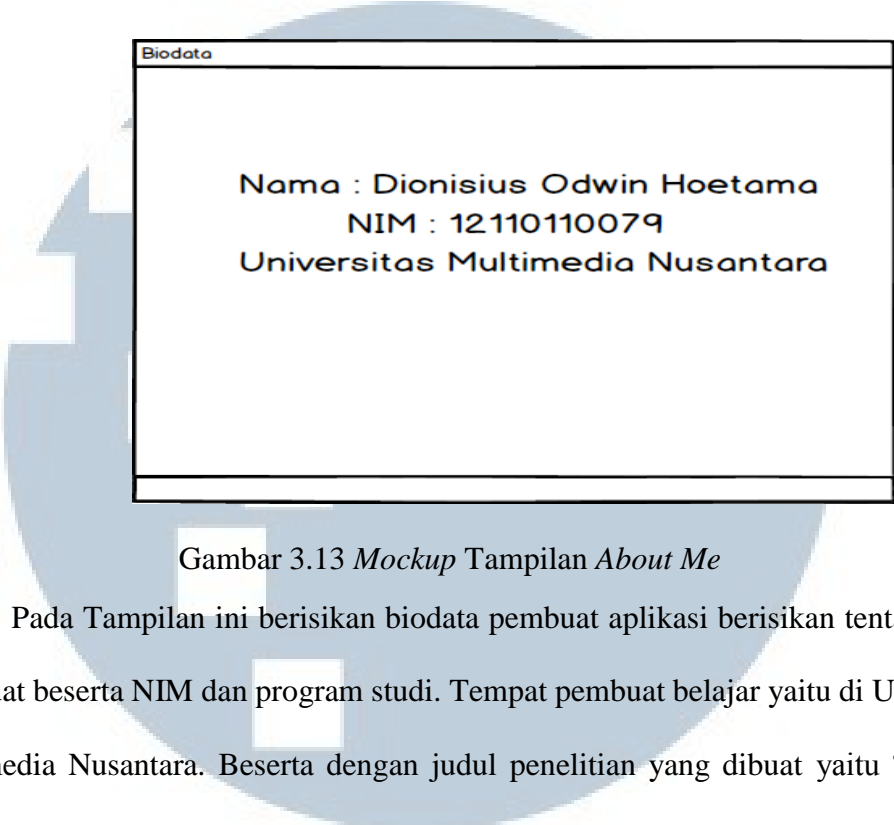
The mockup shows a web form titled "Data Training". At the top, there is a search bar with the label "Jenis Ikan :". Inside the search bar is a dropdown menu with the text "Pilih Jenis" and a downward arrow. To the right of the search bar is a button labeled "Cari". Below the search bar is a table with the following columns: "No.", "Nama", "Jenis", "R", "G", "B", "Waktu", and "Fase K". The table is currently empty.

No.	Nama	Jenis	R	G	B	Waktu	Fase K
-----	------	-------	---	---	---	-------	--------

Gambar 3.12 *Mockup* Form Tampilan Data

Gambar 3.12 *Mockup* Form Tampilan Data

uhui untuk mengeluarkan data-data tersebut. Data yang ditampilkan nama, Jenis, R, G, B, waktu dan Fase Kematian. Tampilan akhir menu biodata pembuat aplikasi yang bernama *About Me*. Tampilan antarmuka tentang biodata seperti nama, NIM, prodi dan lainnya tentang pembuat aplikasi dapat dilihat pada Gambar 3.13 berikut.



Biodata

Nama : Dionisius Odwin Hoetama
NIM : 12110110079
Universitas Multimedia Nusantara

Gambar 3.13 *Mockup Tampilan About Me*

Pada Tampilan ini berisikan biodata pembuat aplikasi berisikan tentang nama pembuat beserta NIM dan program studi. Tempat pembuat belajar yaitu di Universitas Multimedia Nusantara. Beserta dengan judul penelitian yang dibuat yaitu “Rancang Bangun Aplikasi Klasifikasi Waktu Pembusukan Ikan Menggunakan Algoritma KNN Dan *Manhattan Distance*.”

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA