BAB III
PERANCANGAN SISTEM

3.1 Analisis Masalah

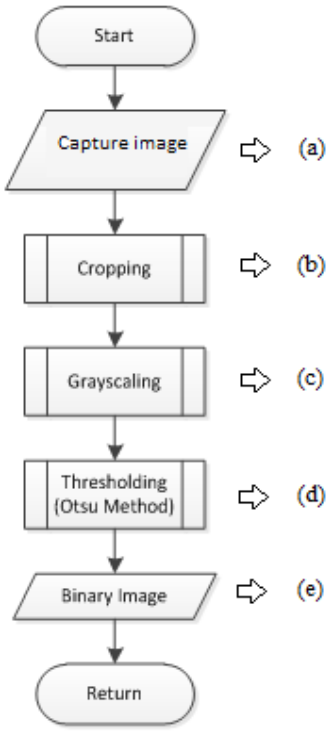
Permasalahan yang diangkat pada penelitian ini ialah menganalisis dan mendiagnosa penyakit ikan kakap melalui tubuh ikan menggunakan program aplikasi sehingga menghasilkan pendiagnosaan penyakit yang optimal. Analisis tersebut dilakukan berdasarkan pengambilan gambar yang mungkin terjadi pada saat pengguna aplikasi melakukan pengambilan gambar pada tubuh ikan secara langsung.

3.2 Analisis Data penyakit

Analisis data penyakit berisi penjelasan tentang penyakit serta contoh gambar yang akan disimpan di database untuk dibandingkan. Tujuannya untuk mendapatkan perbedaan antara yang sehat dan berpenyakit pada ikan. Untuk gambar sample penyakit berada pada lampiran No.1.

3.3 Analisis Pengolahan Citra

Struktur Pengolahan Citra pada program aplikasi terdiri dari tiga buah proses secara berurut yaitu *cropping*, *grayscaling* dan *thresholding*. Tujuan dari modul ini ialah untuk menghasilkan citra yang dapat dikenali oleh smartphone. Urutan proses ditunjukan oleh *flowchart* pada gambar 3.1



Gambar 3.1 Flowchart struktur pengolahan citra

Berikut adalah penjelasan dari setiap proses yang ada pada gambar 3.1

a. Capture Image

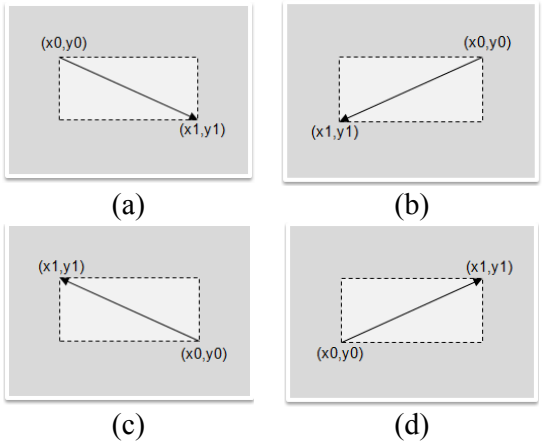
Pada proses ini dilakukan pengambilan gambar pada objek yang akan diproses. Langkah ini adalah awal dari proses proses lainnya karena proses lainnya membutuhkan gambar. Cara pengambilan gambarnya dengan mengarahkan kamera ponsel android ke daerah tubuh ikan yang ingin diidentifikasi.

b. Cropping

Pada proses ini, dilakukan pemotongan terhadap gambar masukan dengan cara menyeleksi area tubuh ikan yang akan dikenali. Proses ini menghasilkan sebuah potongan gambar pada area tubuh ikan yang dilakukan oleh *user*. *Cropping* pada pengolahan citra berarti memotong satu bagian dari citra sehingga diperoleh citra

yang diharapkan. Ukuran pemotongan citra tersebut berubah sesuai dengan ukuran citra yang diambil. *Cropping* dilakukan pada koordinat (*x*,*y*) sampai pada koordinat (*m*,*n*). Oleh karena itu, pertama kali yang harus dilakukan adalah menentukan koordinat-koordinat

tersebut. Ilustrasi *cropping* dapat dilihat pada gambar 3.2 berikut.



Gambar 3.2. Ilustrasi *cropping*: (a) kiri-atas, (b) kanan-atas, (c) kiri-bawah , (d) kanan-bawah

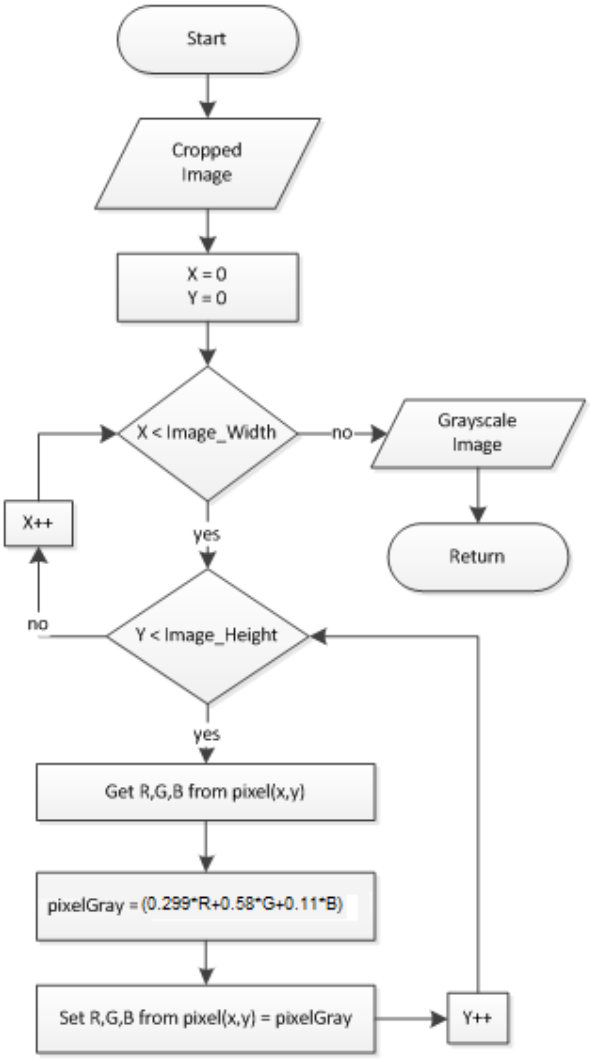
Dari proses *cropping* pada gambar 3.2 di atas akan menghasilkan koordinat (x_m, y_m) dan (x_n, y_n) yang berbeda untuk menentukan nilai serta koordinat awal dari potongan citra yang akan dihasilkan.

c. Grayscaling

Pada proses ini, potongan gambar yang berwarna akan diubah menjadi gambar yang ke abu-abuan (*grayscale*). Transformasi gambar ke bentuk *grayscale* dilakukan dengan mengambil *pixel* dari salah satu gambar yang kemudian dihitung dengan menggunakan persamaan berikut:

$$Grayscale = 0.299R+0.58G+0.11B \quad \text{.....(3.3)}$$

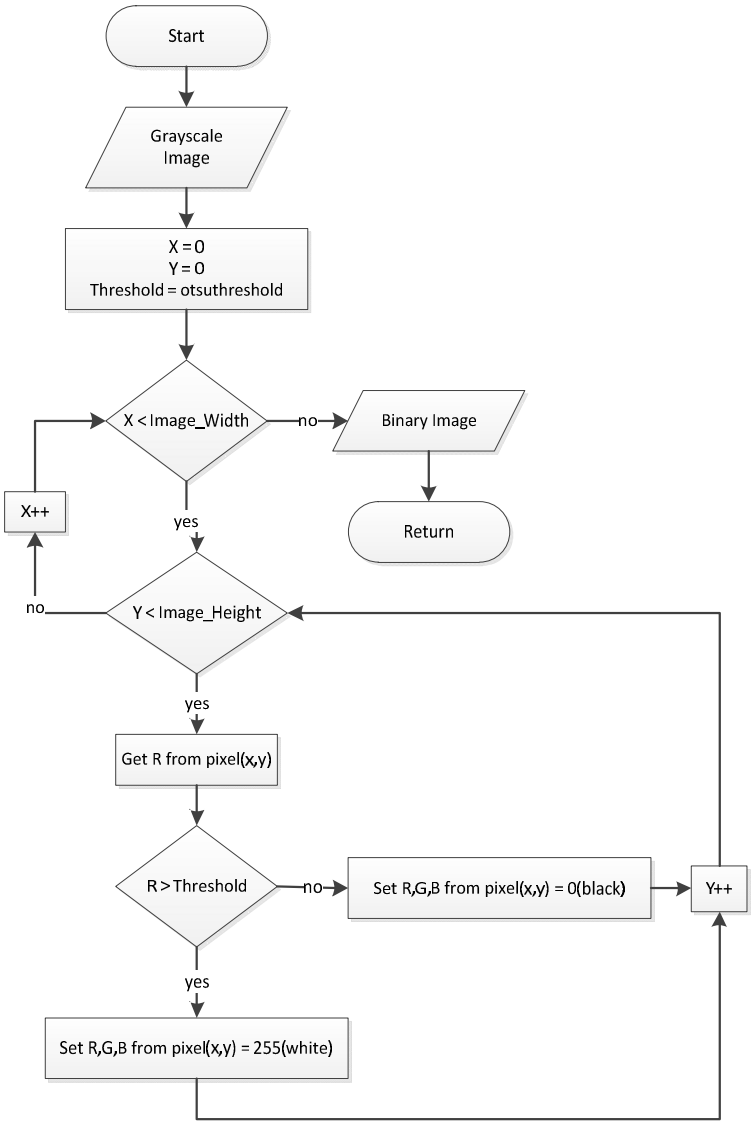
Proses pengubahan citra RGB ke dalam citra *grayscale* dapat dilihat pada gambar 3.3 berikut.



Gambar 3.3 Flowchart grayscaling

d. Thresholding

Thresholding adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk obyek dan *background* dari citra secara jelas. Citra hasil *thresholding* biasanya digunakan lebih lanjut untuk proses pengenalan obyek serta ekstraksi fitur. proses pada gambar 3.4 sebagai berikut:



Gambar 3.4 Flowchart Thresholding

e. Binary Image

Pada tahap ini gambar hasill cropping akan dirubah menjadi dua nilai derajat keabuan yaitu hitam dan putih dengan nilai 0 dan 1. Nilai entri matriks (yang mewakili suatu intensitas cahaya) akan diubah menjadi warna hitam apabila berada di bawah nilai threshold, dan begitu sebaliknya yaitu diubah menjadi warna putih apabila berada di atas nilai threshold tadi

3.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras/*hardware*,analisis perangkat lunak/*software*,analisis pengguna/user.

3.4.1 Analisis kebutuhan Perangkat Keras

Kebutuhan perangkat keras untuk menjalankan aplikasi ini ialah *smartphone* Android yang memiliki beberapa spesifikasi utama sebagai berikut:

1. Sistem operasi Android 4.0 (ICS).
2. Kamera 5 MP (2560x1920 pixels), untuk keperluan pengambilan gambar yang dilakukan secara langsung.
3. Tampilan layar sentuh, untuk keperluan pemotongan gambar yang dilakukan secara manual.
4. Media penyimpanan eksternal, untuk keperluan menyimpan berbagai data yang dihasilkan oleh program aplikasi.
5. Jaringan 2G/3G, untuk menghubungkan program aplikasi dengan layanan server.

3.4.2 Analisis kebutuhan Perangkat Lunak

Perangkat lunak yang dipakai dalam pengembangan program aplikasi, diantaranya:

1. Sistem Operasi Windows 7 Ultimate.
2. Java Development Kit versi 6, untuk keperluan kompilasi kode-kode program.
3. Eclipse IDE 4.2.2, sebagai perangkat lunak yang digunakan untuk membangun aplikasi android.
4. Android SDK, untuk pengembangan aplikasi Android
5. Android Development Tool, sebagai plugin untuk mengintegrasikan Android SDK pada eclipse.

6. Filezilla FTP Client, untuk mengakses server.

3.4.3 Analisis Pengguna

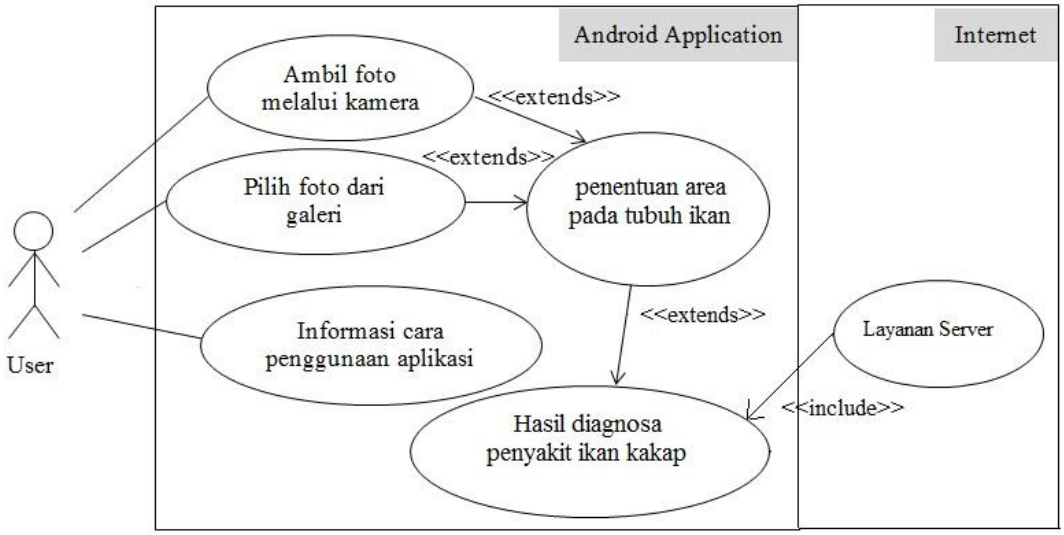
Aplikasi yang akan dibangun ditujukan bagi para pengusaha yang membudidayakan ikan kakapdapat mengidentifikasi penyakit ikan kakap menggunakan smartphone android. Aplikasi ini juga dapat difungsikan sebagai pembelajaran tentang penyakit ikan kakap serta penanggulangannya.

3.5 Analisis Kebutuhan Fungsional

Kebutuhan fungsional yang digunakan meliputi beberapa diagram UML, diantaranya: diagram *use case*, diagram *activity*, diagram *class* dan diagram *sequence*. Diagram tersebut digunakan untuk mempermudah proses pengembangan aplikasi yang dikembangkan menggunakan pemrograman berorientasi objek.

3.5.1 Diagram Use Case

Pada sistem aplikasi yang dirancang terdapat lima buah *uses case* dan *user* sebagai aktor tunggal yang berperan dalam sistem. Kata kunci *extends* menjelaskan bahwa *use case* tersebut merupakan perluasan dari *use case* sebelumnya.



Gambar 3.5 Diagram Use Case

Penjelasan diagram *use case* diatas menggunakan tabel spesifikasi naratif sebagai berikut:

Tabel 3.1. Spesifikasi naratif untuk *use case* ambil foto melalui kamera

Nama *Use Case*	**Ambil foto melalui kamera**
Deskripsi Singkat	*Use case* ini memungkinkan pengguna untuk melakukan pengambilan foto secara langsung menggunakan kamera.
Aktor	Pengguna aplikasi.
Pra Kondisi	Aplikasi telah dijalankan.
Tindakan Utama	Pengguna mengambil foto dengan cara memilih menu Ambil Foto yang terdapat di aplikasi.
Pasca Kondisi	Foto ditampilkan di layar.

Tabel 3.2. Spesifikasi naratif untuk *use case* pilih foto dari galeri

Nama *Use Case*	**Pilih foto dari galeri**
Deskripsi Singkat	*Use case* ini memungkinkan pengguna untuk memilih foto yang sudah tersimpan di *external memory* melalui galeri.
Aktor	Pengguna aplikasi.
Pra Kondisi	Aplikasi telah dijalankan.
Tindakan Utama	Pengguna memilih foto yang sudah tersimpan di *external memory* dengan cara memilih menu Pilih Foto dari Gallery yang terdapat di aplikasi.
Pasca Kondisi	Foto ditampilkan di layar.

Tabel 3.3. naratif untuk *use case* penentuan area tubuh ikan pada foto

Nama *Use Case*	**Penentuan area pada tubuh ikan**
Deskripsi Singkat	*Use case* ini memungkinkan pengguna untuk menentukan dibagian tubuh mana yang mempunyai perbedaan dari tubuh ikan yang normal.
Aktor	Pengguna aplikasi.
Pra Kondisi	Salah satu *use case* sebelumnya telah dilakukan.
Tindakan Utama	Pengguna menyeleksi area tubuh ikan pada citra yang ingin diproses oleh aplikasi dengan

	memanfaatkan fitur layar sentuh pada perangkat Android.
Pasca Kondisi	Foto ditampilkan di layar.

Tabel 3.4. Spesifikasi naratif untuk *use case* hasil diagnosa pada tubuh ikan

Nama *Use Case*	**Hasil diagnose penyakit ikan**
Deskripsi Singkat	*Use case* ini merupakan perluasan dari *use* case penentuan area pada tubuh ikan. *Use case* akan menampilkan hasil identifikasi dan diagnose pada tubuh ikan dengan menyertakan fungsionalitas dua buah *use case* dari luar system
Aktor	-
Pra Kondisi	*Use case* penentuan area pada foto telah dilakukan.
Tindakan Utama	-
Pasca Kondisi	Hasil identifikasi dan diagnose ditampilkan di layar.

Table 3.5. Spesifikasi naratif untuk *use case* informasi cara penggunaan aplikasi

Nama *Use Case*	**Informasi cara penggunaan Aplikasi**
Deskripsi Singkat	*Use case* ini memungkinkan pengguna untuk mendapatkan informasi mengenai cara

	penggunaan aplikasi.
Aktor	Pengguna aplikasi.
Pra Kondisi	Aplikasi telah dijalankan.
Tindakan Utama	Pengguna memilih menu Bantuan yang terdapat di aplikasi
Pasca Kondisi	Informasi mengenai cara penggunaan aplikasi ditampilkan di layar.

a. Use case scenario

Use case s*cenario* merupakan deskripsi urutan langkah-langkah dalam proses yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Untuk *use case scenario* pada sistem akan di jelaskan pada tabel berikut.

Tabel 3.6. use case *scenario* ambil foto melalui kamera

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu kamera	
	2. Menampilkan sistem kamera
3. Capture gambar	
	4. Menampilkan gambar

Tabel 3.7. use case scenario pilih foto dari galeri

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu galeri	
	2. Menampilkan isi galeri
3. Memilih gambar pada galeri	
	4. Menampilkan gambar

Tabel 3.8. use case scenario penentuan area tubuh ikan pada foto

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu kamera.	
	2. Menampilkan sistem kamera.
3. Capture gambar.	
	4. Menampilkan gambar.
5. Menentukan area yang ingin diproses pada gambar.	
	6. Permintaan konfirmasi pada gambar.
7. Konfirmasi gambar.	

Skenario Alternatif	
1. Memilih menu galeri	
	2. Menampilkan isi galeri
3. Memilih gambar pada galeri	
	4. Menampilkan gambar
5. Menentukan area yang ingin diproses pada gambar	
	6. Permintaan konfirmasi pada gambar.
7. Konfirmasi gambar	

Tabel 3.9. use case scenario hasil diagnosa pada tubuh ikan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu kamera.	
	2. Menampilkan sistem kamera.
3. Capture gambar.	
	4. Menampilkan gambar.

5. Menentukan area yang ingin diproses pada gambar.	
	6. Permintaan konfirmasi pada gambar.
7. Konfirmasi gambar.	
	8. Memproses gambar.
	9. Menampilkan hasil diagnosa gambar.
Skenario Alternatif	
1. Memilih menu galeri	
	2. Menampilkan isi galeri
3. Memilih gambar pada galeri	
	4. Menampilkan gambar
5. Menentukan area yang ingin diproses pada gambar	
	6. Permintaan konfirmasi pada gambar.
7. Konfirmasi gambar	
	8. Memproses gambar

	9. Menampilkan hasil diagnosa gambar.

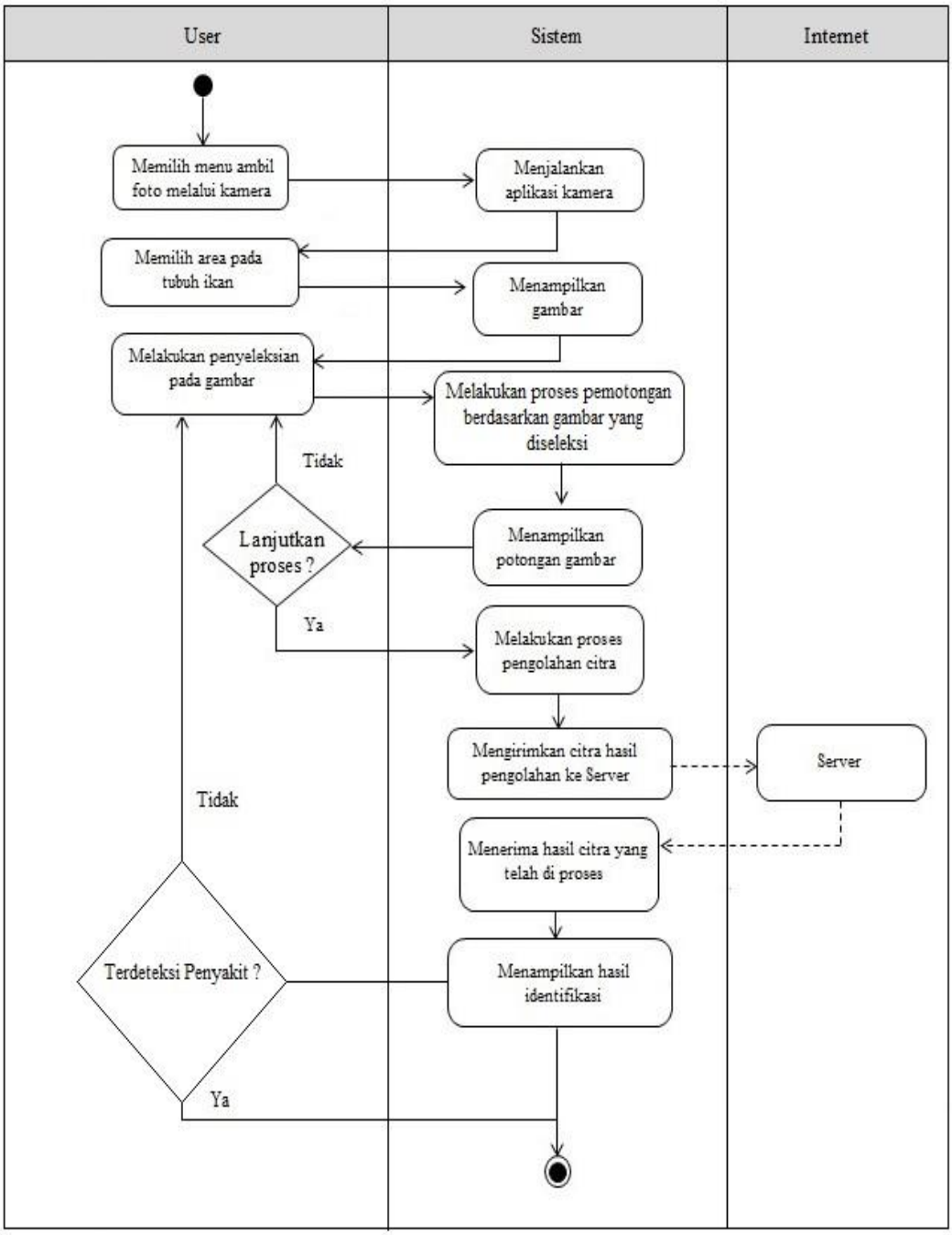
Tabel 3.10. use case scenario informasi cara penggunaan aplikasi

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu cara penggunaan aplikasi	
	2. Menampilkan cara penggunaan aplikasi.

3.5.2 Diagram Activity

Diagram *activity* digunakan untuk menggambarkan alur rangkaian proses yang terjadi di dalam sistem aplikasi serta menjelaskan bagaimana sebuah *use case* terjadi atau dipicu dari *use case* yang lain. Masing-masing diagram aktifitas dapat dilihat pada gambar 3.6 sampai dengan 3.8.

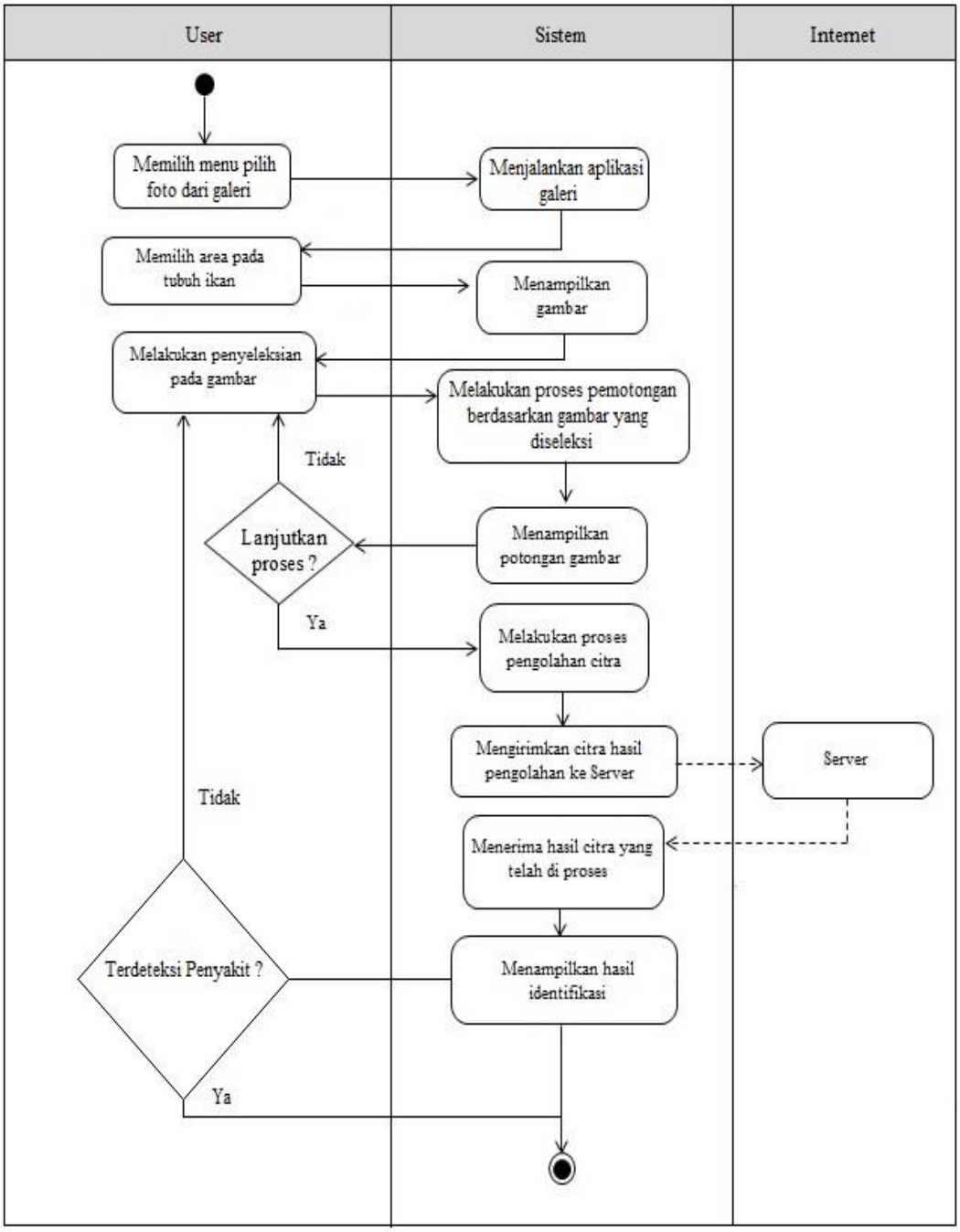
Gambar 3.6 memperlihatkan aktifitas yang terjadi antara *user* dan sistem pada saat *user* memilih menu ambil foto melalui kamera. Gambar 3.7 memperlihatkan aktifitas yang terjadi antara *user* dan sistem pada saat *user* memilih menu pilih foto dari galeri. Kedua diagram tersebut pada umumnya sama, perbedaannya hanya ada pada menu pengambilan gambarannya. Untuk itu, penjelasan mengenai gambar 3.7 akan mengacu pada penjelasan gambar 3.6.



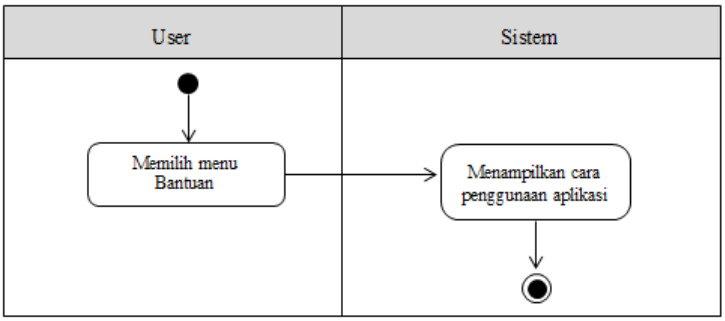
Gambar 3.6. Diagram Activity ambil foto melalui kamera

Diagram *activity* pada gambar di atas menggambarkan aktifitas yang dilakukan *user* pada saat memilih menu ambil foto melalui kamera sampai dengan sistem menampilkan hasil identifikasi. Penjelasan mengenali aktifitas yang terjadi:

1. Aktifitas dimulai dari *user* memilih menu ambil foto melalui kamera;
2. Sistem aplikasi selanjutnya akan memanggil aplikasi kamera yang kemudian memicu *use case* ambil foto melalui kamera.
3. Foto hasil pengambilan tersebut akan ditampilkan oleh sistem yang kemudian memicu *use case* berikutnya yaitu penyeleksian area pada foto.
4. Sistem kemudian melakukan proses pemotongan gambar berdasarkan area yang diseleksi.
5. Pada saat sistem melakukan aktifitas menampilkan potongan gambar, sistem melakukan pengkondisian agar user dapat melakukan kembali aktifitas penyeleksian area pada tubuh ikan
6. Jika user memilih untuk melanjutkan proses, maka sistem akan melakukan prosedur pengolahan citra.
7. selanjutnya hasil citraakan ditampilkan di layar.



Gambar 3.7. Diagram Activity *memilih foto dari galeri*

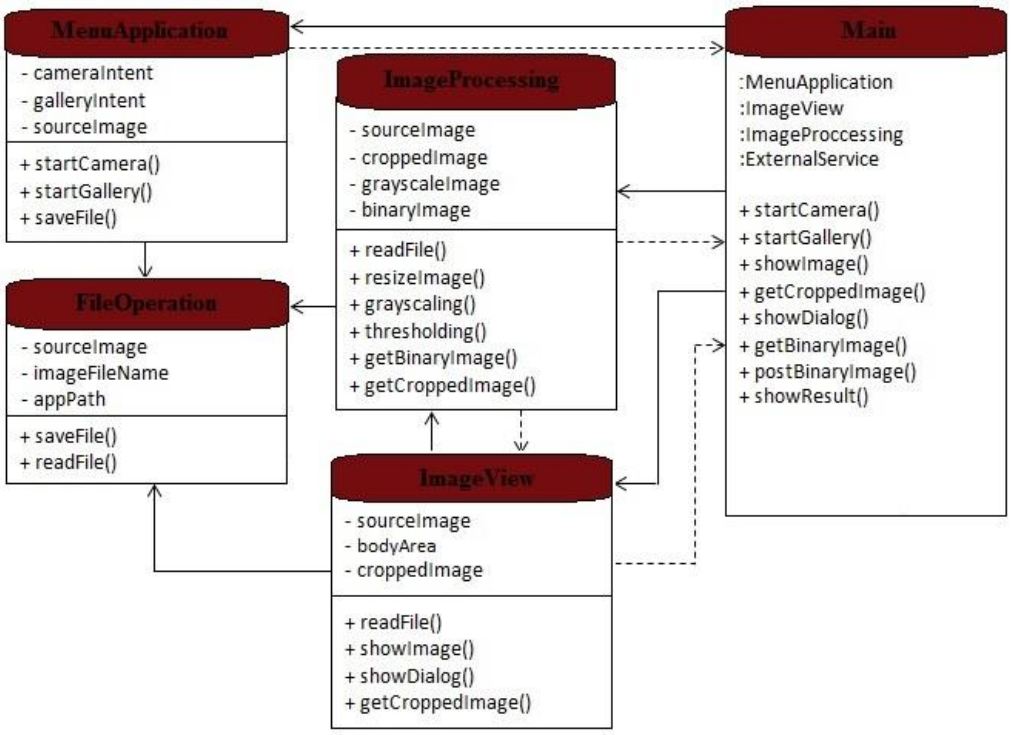


Gambar 3.8. Diagram Activity *lihat cara penggunaan aplikasi*

Diagram *activity* di atas menggambarkan aktifitas yang dilakukan *user* untuk memperoleh informasi mengenai cara penggunaan aplikasi.

3.5.3 Diagram *Class*

Diagram *Class* digunakan untuk menggambarkan secara abstrak struktur dari aplikasi yang akan dibangun, *class-class* yang terlibat, serta hubungan antar *class* untuk saling berkomunikasi satu sama lain. Struktur *class* dari aplikasi yang akan dibangun dapat dilihat pada gambar 3.9.

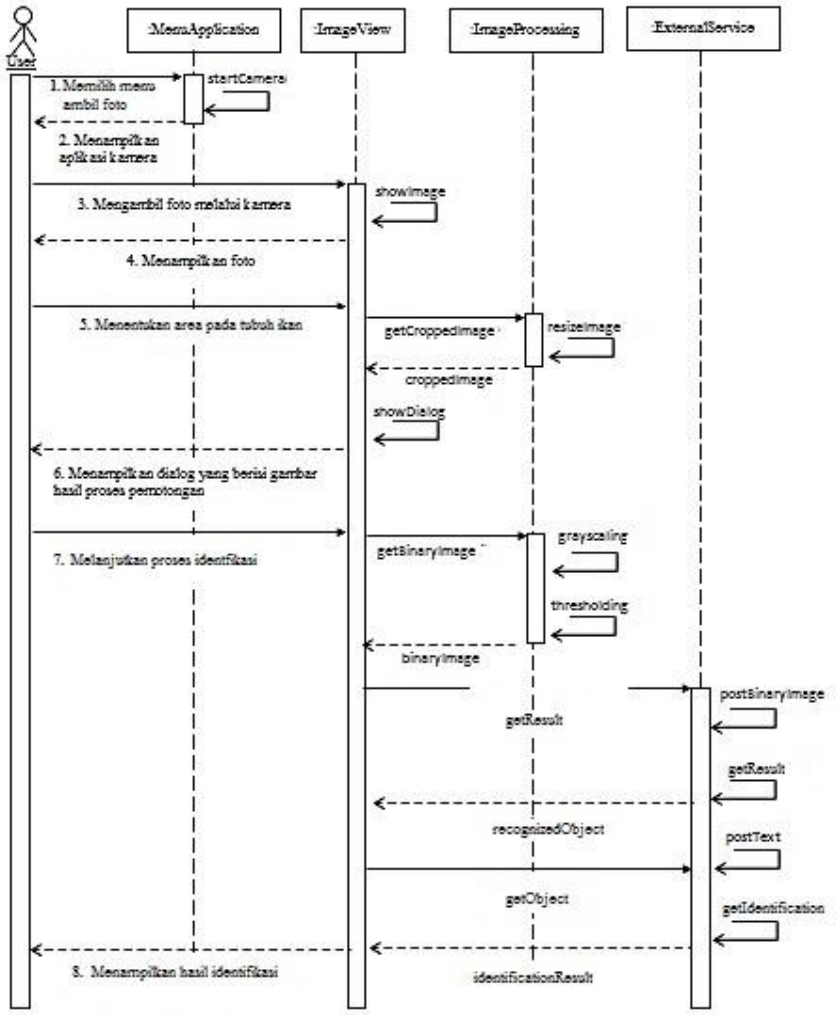


Gambar 3.9. Diagram Class

Aplikasi Android yang akan dibangun terdiri dari lima buah *class*, dengan *class* Main sebagai *class* utamanya. Pada *class* Main terdapat empat objek yang merupakan interpretasi dari *class* lain yang terlibat. Pada diagram *class* diatas terlihat dua jenis garis yang menghubungkan antara *class* yang satu dengan *class* lain. Garis penuh menunjukkan pertukaran pesan antar objek, sedangkan garis putus-putus menunjukkan respon dari objek yang dipanggil.

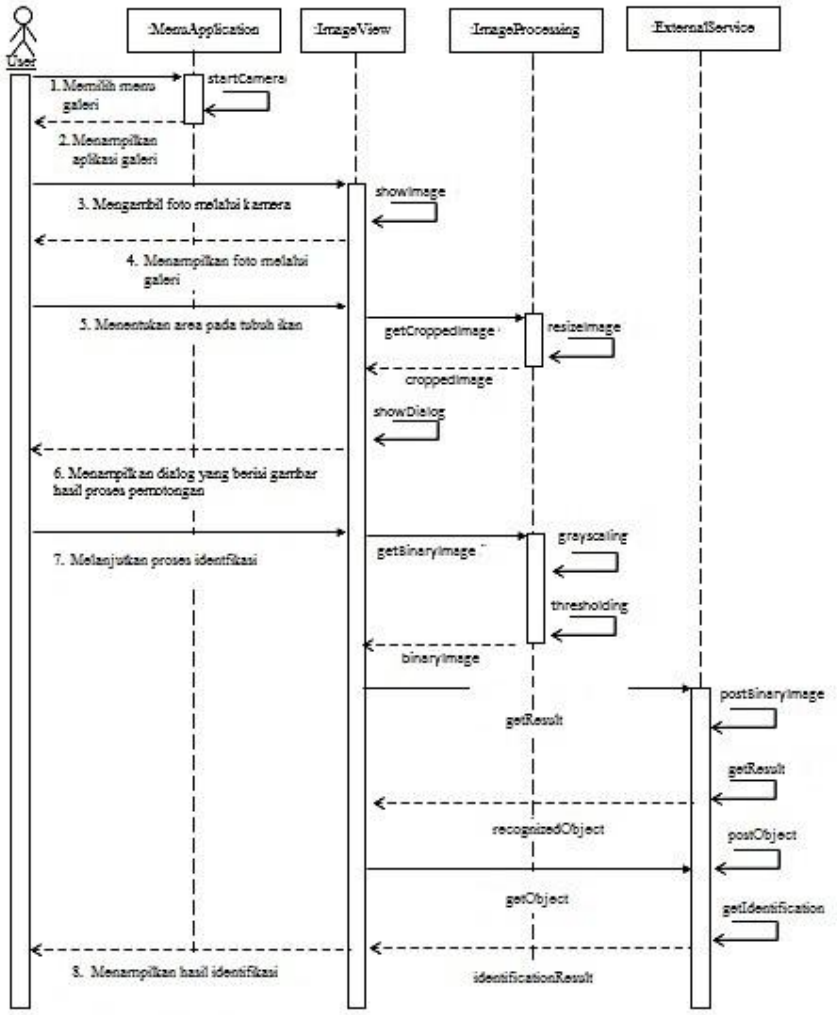
3.5.4 Diagram *Sequence*

Diagram *sequence* menjabarkan urutan aktifitas yang terjadi di dalam sistem. Diagram ini menunjukkan pengiriman pesan-pesan yang melewati objek yang terlibat di dalam sistem serta tahapan langkah yang harus dilalui untuk mencapai *use case* tertentu. Diagram *sequence* dari aplikasi yang akan dibuat, dapat dilihat pada gambar 3.10. sampai dengan gambar 3.12.



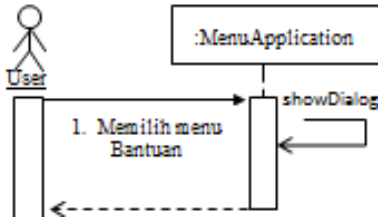
Gambar 3.10. Diagram Sequence *ambil foto melalui kamera*

Pada diagram *sequence* diatas, terdapat delapan aktifitas yang terjadi hingga *user* memperoleh hasil Identifikasi penyakit pada ikan kakap. *User* berinteraksi secara langsung dengan objek MenuAplication dan ImageView. Pengiriman pesan antar objek lebih banyak melibatkan objek ImageView yang merupakan antarmuka *user* dengan program aplikasi.



Gambar 3.11. Diagram Sequence *pilih foto dari galeri*

Pada diagram *sequence* diatas, sebagian besar aktifitas yang terjadi sama dengan diagram *sequence* pada gambar 3.10. Perbedaan hanya ada pada aktifitas ke-1 sampai dengan aktifitas ke-3. Untuk itu penjelasan mengenai diagram ini akan mengacu pada penjelasan gambar 3.10.



Gambar 3.12. Diagram Sequence *cara penggunaan aplikasi*

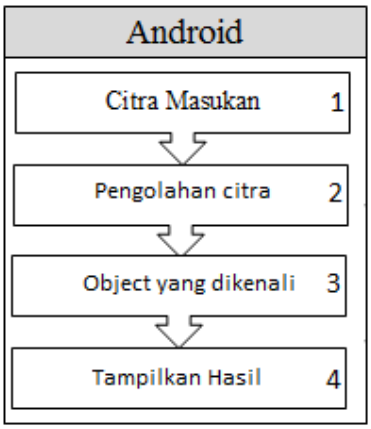
Pada diagram *sequence* di atas terdapat dua aktifitas yang terjadi hingga *user* memperoleh informasi mengenai cara penggunaan aplikasi. Objek yang terlibat ialah MenuAplication yang merupakan antarmuka *user* dengan program aplikasi pada saat pertamakali aplikasi dijalankan.

3.6 Perancangan Sistem

Sistem aplikasi android yang akan dirancang memiliki fungsi utama untuk melakukan proses pengolahan citra. Tujuan dari pengolahan citra yang dikerjakan oleh aplikasi ini ialah untuk menghasilkan citra yang bisa mengidentifikasi masalah tentang penyakit ikan.

3.6.1 Rancangan Sistem Aplikasi Android

Secara umumnya, rancangan sistem aplikasi yang akan dibangun dapat dilihat pada gambar 3.13.



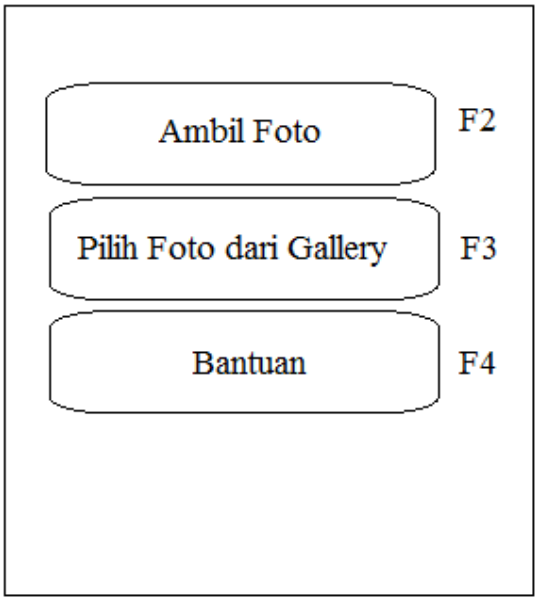
Gambar 3.13. Diagram blok rancangan sistem aplikasi Android

Penjelasan rancangan sistem aplikasi Android:

1. Citra masukan merupakan citra area tubuh ikan yang akan diproses oleh program aplikasi. Citra ini diperoleh dari hasil pengambilan foto secara langsung atau bisa juga berasal dari galeri yang ada pada *smartphone* Android.
2. Pengolahan citra pada program aplikasi Android akan mengubah citra masukan menjadi sebuah citra gambar berwarna hitam dengan latar belakang berwarna putih.
3. Setelah citra berhasil dikenali, hasil pengenalan tersebut kemudian akan di bandingkan dengan yang ada di database
4. Jika mendekati dengan hasil yang berada di database maka aplikasi akan memunculkan gambar yang serupa
5. Tampilan akhir pada program aplikasi berupa hasil identifikasi penyakit serta cara penanggulangannya.

3.6.2 Rancangan Antarmuka

Sistem aplikasi yang akan dibuat mempunyai dua tampilan, yaitu tampilan Menu Utama dan tampilan *Image View*. Tampilan Menu Utama digunakan untuk menentukan sumber foto yang ingin di idenrtifikasi sedangkan *Image View* digunakan untuk menampilkan foto tersebut.



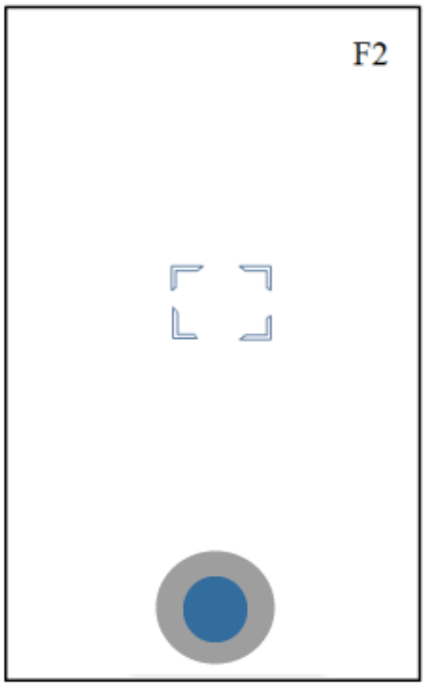
Gambar 3.14. Antarmuka aplikasi Android Form 1.

Keterangan:

Pada saat tombol Ambil Foto ditekan, akan memanggil aplikasi kamera. Seperti yang terlihat pada tampilan F2 .

Pada saat tombol Pilih Foto dari Gallery di tekan, akan memanggil aplikasi gallery. Seperti yang terlihat pada tampilan F3.

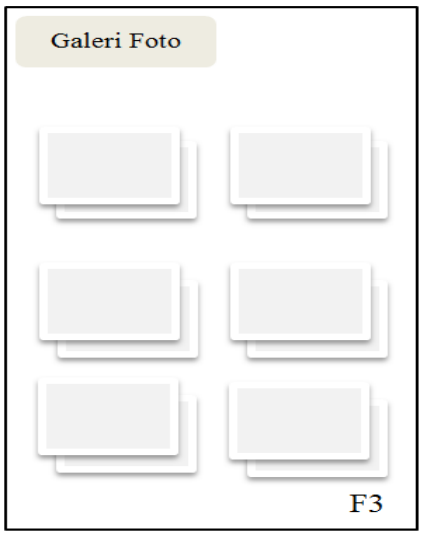
Pada saat tombol Bantuan di tekan, akan memanggil membuka dialog cara penggunaan aplikasi, seperti yang terlihat pada tampilan F4.



Keterangan:

Ketika tombol *icon* kamera ditekan, maka foto akan disimpan di *memory external* dan foto tersebut akan ditampilkan pada tampilan F5

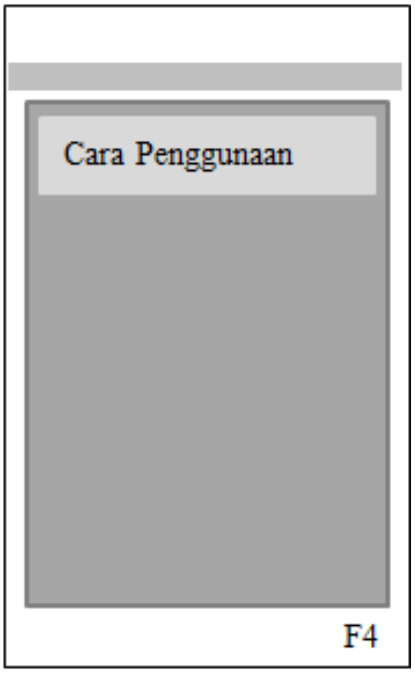
Gambar 3.15. Antarmuka aplikasi Android Form 2



Keterangan:

Ketika salah satu foto sudah dipilih, maka foto tersebut akan ditampilkan pada tampilan F5.

Gambar 3.16. Antarmuka aplikasi Android Form 3



Keterangan:

Form ini berfungsi sebagai petunjuk cara penggunaan aplikasi.

Gambar 3.17. Antarmuka aplikasi Android Form 4



Keterangan:

Pada tampilan ini user menentukan sendiri area yang ingin dikenali, kemudian hasil pengenalan dan diagnose penyakit ikan tersebut akan terlihat seperti pada tampilan F6.

Gambar 3.18. Antarmuka aplikasi Android Form 5



Keterangan:

Ketika tombol kembali pada perangkat ditekan, maka tampilan akan kembali ke F1.

Gambar 3.19. Antarmuka aplikasi Android Form 6

3.7 Data Ikan

Setiap gambar yang diupload mempunyai nilai tersendiri yang akan Diband-ingkan kepada server. Berikut ini ada nilai dari setiap data ikan yang ada.

A. Penyakit insang

Penyakit pada insang mempunyai index similarity dengan rentang 72-74

Tabel 3.11 index similarity insang yang sakit

No	Penyakit	*Index Similarity*
1	Insang sakit 1	72
2	Insang sakit 2	72.5
3	Insang sakit 3	72.8

4	Insang sakit 4	73.4
5	Insang sakit 5	73.6
6	Insang sakit 6	73.9
7	Insang sakit 7	72.4
8	Insang sakit 8	72.9
9	Insang sakit 9	73.8
10	Insang sakit 10	72.6
Rata - rata		72.99

B. Penyakit badan

Penyakit pada badan mempunyai *index similarity* dengan rentang 70-71.9

Tabel 3.12 index similarity badan yang sakit

No	Penyakit	*Index Similarity*
1	Badan sakit 1	71.5
2	Badan sakit 2	70.2
3	Badan sakit 3	70
4	Badan sakit 4	70.6
5	Badan sakit 5	71.5
6	Badan sakit 6	71.8
7	Badan sakit 7	71.6

8	Badan sakit 8	71.3
9	Badan sakit 9	71
10	Badan sakit 10	70.9
Rata - rata		71.04

C. Penyakit ekor

Penyakit pada ekor mempunyai index similarity dengan rentang 68-69.9

Tabel 3.13 index similarity ekor yang sakit.

No	Penyakit	*Index Similarity*
1	Ekor sakit 1	68
2	Ekor sakit 2	68.7
3	Ekor sakit 3	68.4
4	Ekor sakit 4	68.6
5	Ekor sakit 5	69.2
6	Ekor sakit 6	68.8
7	Ekor sakit 7	68.1
8	Ekor sakit 8	68.5
9	Ekor sakit 9	69.4
10	Ekor sakit 10	69.3
Rata - rata		68.7

D. Penyakit sirip

Penyakit pada sirip mempunyai *index similarity* dengan rentang 65-66.9

Tabel 3.14 index similarity sirip sakit.

No	Penyakit	*Index Similarity*
1	Sirip sakit 1	65
2	Sirip sakit 2	65.2
3	Sirip sakit 3	65.8
4	Sirip sakit 4	66.4
5	Sirip sakit 5	66
6	Sirip sakit 6	65.4
7	Sirip sakit 7	65.1
8	Sirip sakit 8	65.3
9	Sirip sakit 9	65.2
10	Sirip sakit 10	66.2
Rata - rata		65.56

E. Penyakit anus

Penyakit pada anus mempunyai *index similarity* dengan rentang 63-64.9

Tabel 3.15 index similarity anus sakit

No	Penyakit	*Index Similarity*
1	Anus sakit 1	63

2	Anus sakit 2	63.2
3	Anus sakit 3	64
4	Anus sakit 4	64.5
5	Anus sakit 5	63.6
6	Anus sakit 6	63.9
7	Anus sakit 7	64.7
8	Anus sakit 8	63.4
9	Anus sakit 9	63.5
10	Anus sakit 10	64.2
Rata - rata		63.8

Dari beberapa index similarity diatas nantinya akan dibandingkan dengan data gambar yang akan diupload ke server, setelah itu sistem akan mengunci penyakit ikan yang diderita untuk sementara. Selanjutnya sistem akan membandingkan dengan database gambar kondisi yang sehat.