BAB III
ANALIS DAN PERANCANGAN

3.1. Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan. Analisis dapat juga diartikan memahami sistem pemikiran yang kompleks dengan memecahnya ke dalam unsur- unsur yang lebih sederhana sehingga hubungan antar unsur-unsur itu menjadi jelas.

3.1.1. Analisis sistem yang sedang berjalan

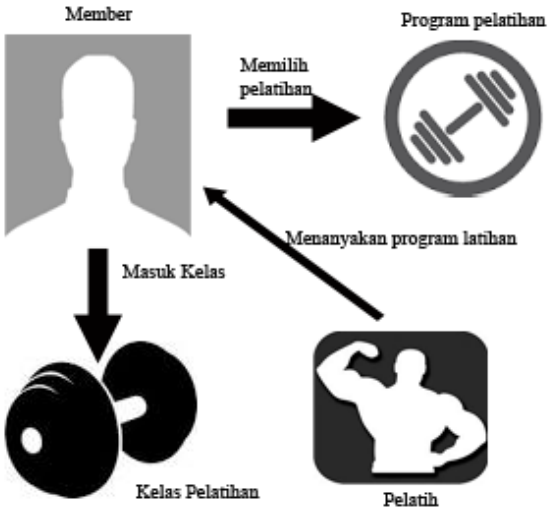
Analisis sistem yang sedang berjalan akan dijelaskan dibawah ini, mulai dari Gambar 3.1 sampai dengan Gambar 3.6

1. Member mendaftar ke admin Rai fitness Purwakarta



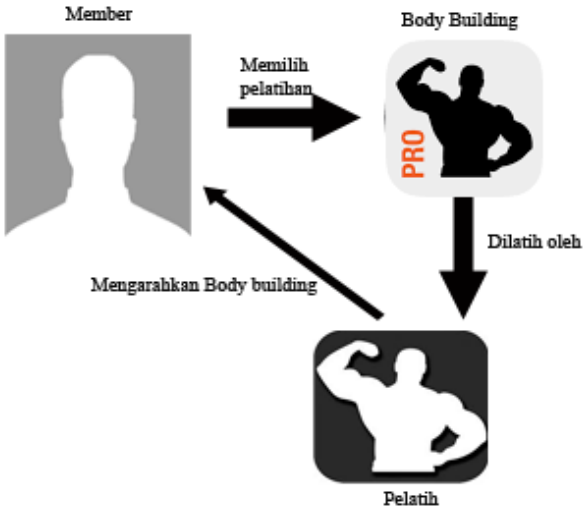
Gambar 3.1 Member mendaftar

2. Member masuk kelas pelatihan, dan pelatih akan bertanya ingin bentuk tubuh yang seperti apa Fitness model atau Body building.



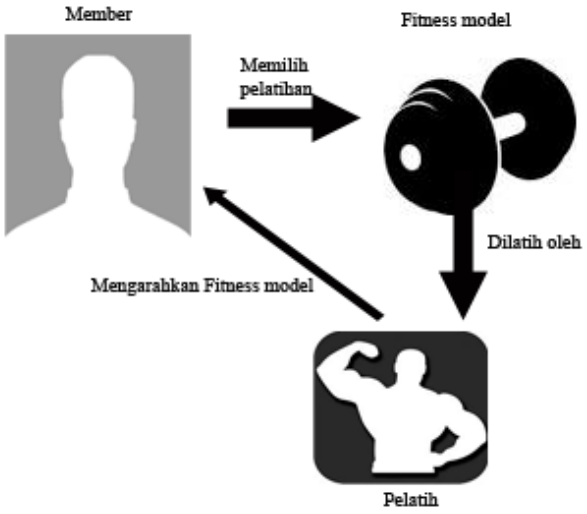
Gambar 3.2 Masuk kelas

3. Jika member memilih Body building maka pelatih akan mengarahkan latihan apa saja yang harus dilakukan.



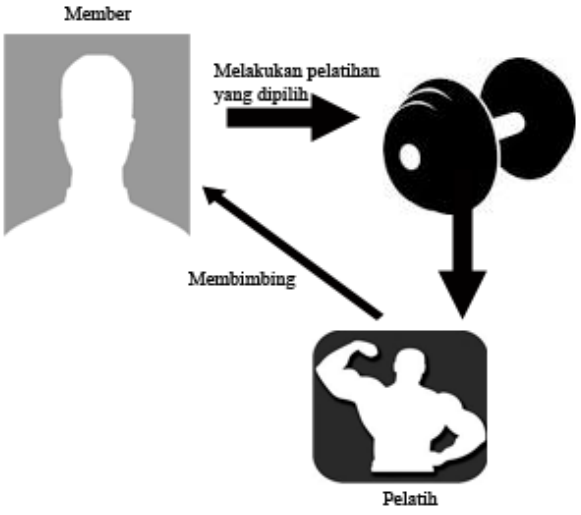
Gambar 3.3 Body building

4. Jika member memilih Fitness model maka pelatih akan mengarahkan latihan apa saja yang harus dilakukan.



Gambar 3.4 Fitness model

5. Kemudian member melakukan latihan sesuai yang dipilih oleh member dengan arahan dari pelatih.



Gambar 3.5 Arahan pelatih

6. Member mendapatkan hasil yang diinginkan pada bentuk tubuhnya.



Gambar 3.6 Bentuk tubuh

3.1.2. Analisis Masalah

Analisis masalah pada sub-bab ini adalah penjabaran masalah apa saja yang ada di Rai Fitness Purwakarta sebelum dibangunnya aplikasi hasil dari pengolahan tubuh ini. Oleh karena itu analisis masalah yang terdapat di Rai Fitness meliputi hal-hal sebagai berikut :

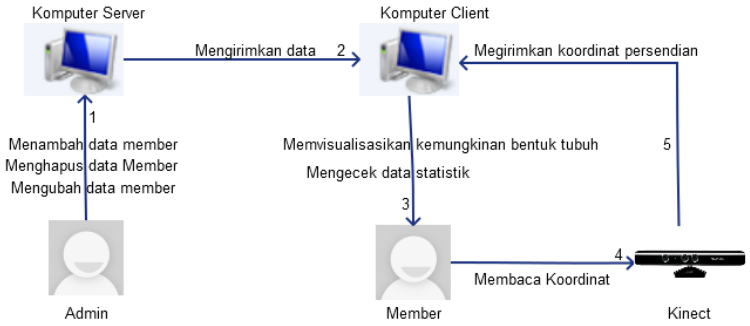
1. Belum tersedianya sarana untuk memvisualisasikan kemungkinan bentuk tubuh yang dapat dicapai.
2. Belum tersedianya sarana untuk mengecek pertumbuhan massa otot dengan menampilkan foto diri.
3. Belum tersedianya sarana untuk mengetahui saran pelatihan dan pola diet berdasarkan bentuk tubuh yang dipilih.

3.1.3. Analisis sistem yang akan dibangun

Aplikasi yang akan dibangun berupa Aplikasi desktop dan aktor utamanya ialah member. Aplikasi ini dibangun sebagai sarana untuk mempermudah pelatih memberikan gambaran kepada member tentang bentuk tubuh yang akan dicapainya, dan juga member dapat dengan mudah untuk mengecek perubahan pada massa ototnya karena disediakanya menu untuk melihat data foto bentuk tubuhnya yang disimpan pada setiap kali latihan, member juga dimudahkan untuk mengetahui pola diet dan saran pelatihan yang baik untuk mendapatkan bentuk tubuh yang diharapkan secara

optimal. Analisis sistem yang akan dibangun dijelaskan oleh Gambar 3.7 berikut penjelasan dari analisis sistem yang akan dibangun.

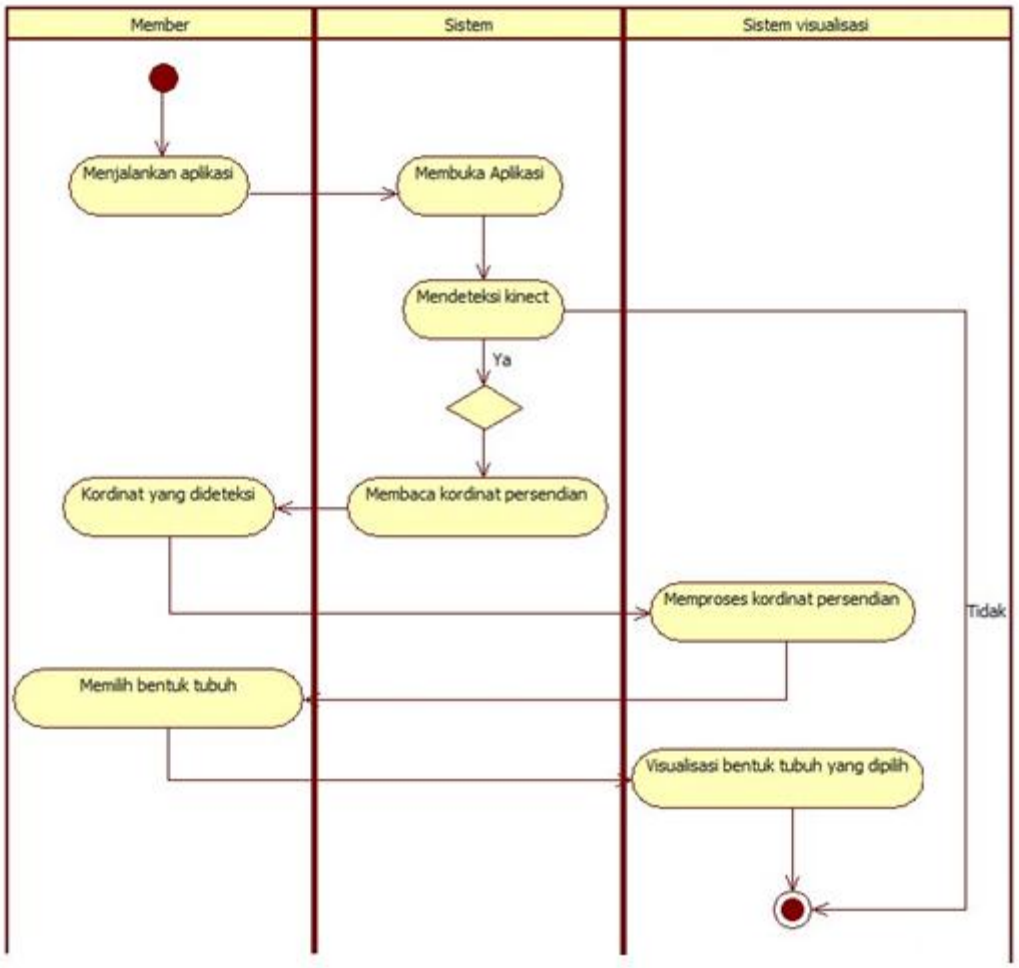
1. Admin bisa melakukan tambah, ubah dan hapus data member yang dikelola oleh admin pada komputer server.
2. Komputer server mengirimkan data yang dibutuhkan oleh komputer client.
3. Member bisa memvisualisasikan berbagai bentuk tubuh menggunakan sensor kinect.
4. Kinect membaca koordinat.
5. Kinect mengirimkan koordinat persendian.
6. Member dapat melakukan foto diri di menu data statistik



Gambar 3.7 Analisis sistem yang akan dibangun

3.1.4. Analisis Arsitektur Sistem

Pada arsitektur sistem yang akan dibangun terdiri 2 sistem yang akan diinstall di komputer server dan komputer client, Sistem yang dibangun di komputer server dikelola oleh admin hanya dapat menambah, meghapus dan mengubah data, sedangkan sistem yang dibangun di komputer client yang digunakan oleh member, dapat memvisualisaikan kemungkinan berbagai bentuk tubuh, menampilkan saran pelatihan dan pola diet dari bentuk tubuh yang dipilih. Komputer client juga dapat melakukan foto diri dan menyimpan kedalam sistem didalam menu data statistik member. alur sistem yang akan dibangun dapat dilihat pada gambar 3.8



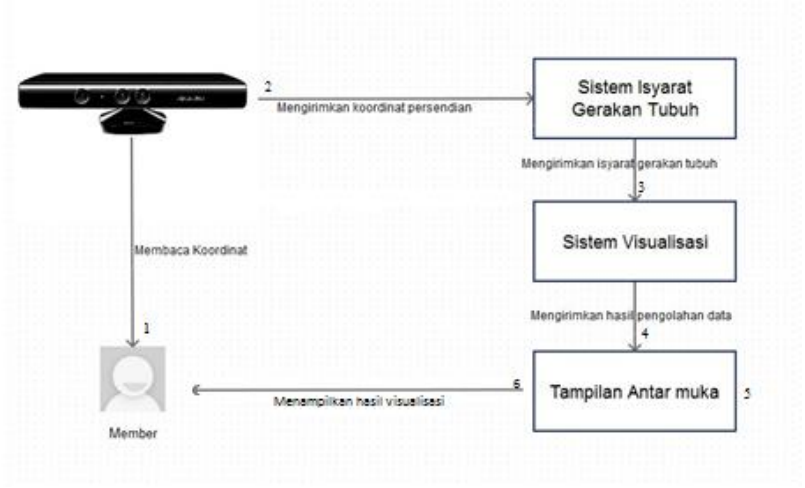
Gambar 3.8 Alur sistem

Berikut penjelasan tentang arsitektur sistem sisi client:

1. Member membuka aplikasi.
2. Sistem Mendeteksi kinect.
3. Kinect membaca kordinat persendian dari sensor kedalaman yang diproyeksikan oleh kamera infrared sebagai inputan.
4. Kinect mengirimkan koordinat persendian yang telah diubah menjadi data skeletal ke sistem isyarat gerakan tubuh.

5. Kemudian member diperkenankan memilih berbagai macam kemungkinan bentuk tubuh lalu setelah memilih kemungkinan bentuk tubuh kemudian sistem akan mendefinisikan titik skeletal mana saja yang digunakan.
6. Sistem visualisasi akan mencari kemungkinan bentuk tubuh yang dipilih kemudian mendeteksi titik titik sendi, jika titik sendi terdeteksi maka.
7. Member dapat melihat hasil visualisasi didalam tampilan antar muka.

Gambaran arsitektur sistem dapat dilihat pada gambar 3.9



Gambar 3.9 Arsitektur sistem

Sistem yang akan dibangun disini ialah sistem yang dapat mendeteksi isyarat tubuh manusia yang kemudian dapat mengirim perintah pada program dan menampilkan hasil visualisasi. Sistem ini terdiri dari sebuah Kinect sebagai sensor gerak serta sebuah komputer untuk pemrosesan data dan interface untuk program. Kinect digerakkan dengan berbagai gesture yang telah dirancang Prinsip kerja sistem ini dimulai isyarat yang diberikan oleh pengguna pada sensor Kinect. Kemudian komputer melakukan pemrosesan, dan sensor kinect yang dihubungkan lewat kabel USB menterjemahkan skeleton kemudian menampilkan hasil visualisasi.

3.1.5. Analisis Isyarat Gerak Tubuh (Gestures)

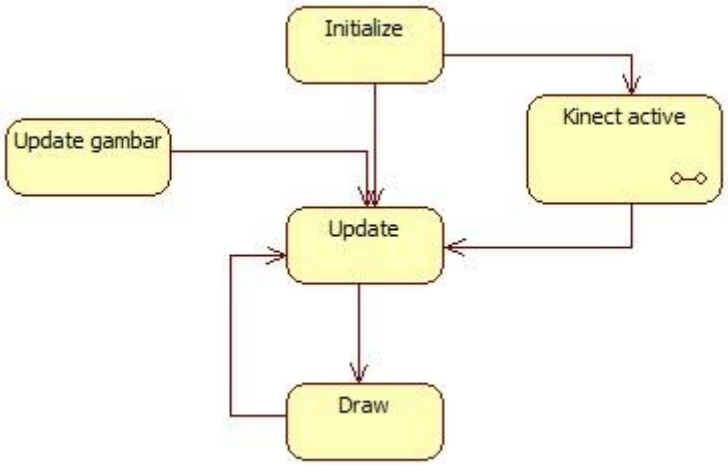
Berikut analisis isyarat gerakan tubuh dijelakan dibawah ini:

3.1.5.1. Library

Library yang digunakan disini yaitu library XNA versi 4.0 didalam library XNA terdapat 3 bagian penting yaitu

a. Initialize yang berisi tampilan tampilan windows dan menu menu
b. Update yang berisi hasil pembacaan sensor kinect per frame yang di deteksi.
c. Draw yang berisi gambar,texture dan text.

Untuk lebih jelasnya dapat dilihat pada gambar 3.10



Gambar 3.10 Diagram alur library

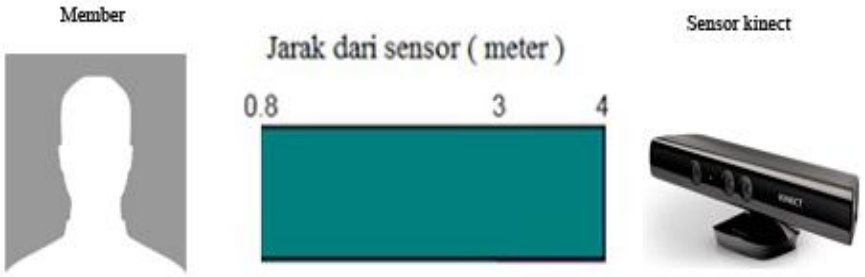
Berikut penjelasan tentang alur library

1. Pada tahap Initialize sistem akan mengaktifkan program, memunculkan tampilan menu, kemudian mengaktifkan kinect lalu mendeteksi koordinat persendian.

   Berikut tahapan yang dilakukan dalam initialize :

   a. Member berdiri di depan sensor dengan jarak sekitar satu sampai empat meter agar rangka tubuh (skeleton) dapat terbaca. Rentang jarak yang dijinkan agar rangka tubuh operator dapat terdeteksi dengan baik antara 0.8 meter

sampai dengan empat meter, penjelasan rentang deteksi sensor dapat dilihat pada Gambar 3.11



Gambar 3.11 Jarak antara sensor dan member

2. Pada tahap Update disini sistem menunggu perubahan / pergerakan koordinat yang dibaca oleh sensor kinect, pergerakan yang dilakukan akan di update setiap frame yang terdeteksi, kemudian kinect akan mengirimkan perubahan koordinat kedalam tahapan draw.

   Berikut tahapan yang dilakukan dalam tahapan update:

   a. Mendefinisikan sendi-sendi yang akan digunakan. Pada penelitian ini digunakan enam persendian meliputi:
      1. Spine joint
      2. Shoulder center joint
      3. Shoulder right joint
      4. Shoulder left joint
      5. Elbow right joint
      6. Elbow left joint

   Shoulder center joint, Spine joint, shoulder right, shoulder left joint digunakan menentukan tinggi gambar yang akan digunakan. Shoulder Right joint, Shoulder Left Joint digunakan untuk menentukan lebar gambar yang akan digunakan.

Berikut sepenggal source code yang digunakan dalam mendefinisikan persendian di bagian tubuh dan di bagian lengan, yang digunakan dalam program.

A.Bagian tubuh

Source code

//this.DrawBody(skeleton.Joints,
JointType.ShoulderCenter, //Pusat persendian
JointType.Spine, // Tulang tengah
JointType.ShoulderLeft, //Bahu kiri
JointType.ShoulderRight);// Bahu kanan

B.Bagian lengan

Source code

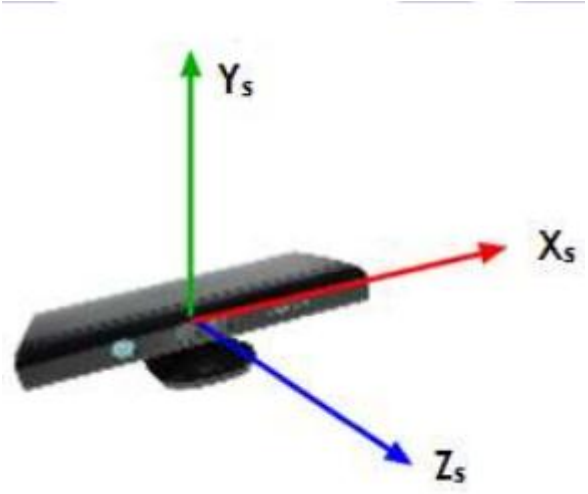
//this.DrawHand(shoulderLeftTexture,
skeleton.Joints,
JointType.ShoulderRight, //Bahu kanan
JointType.ElbowRight, // siku kanan
jointType. ElbowLeft); //siku kiri
break;

3. Pada tahap Draw disini sistem akan menampilkan hasil yang dikirimkan oleh tahap update berupa Gambar, Di tahap ini sistem akan terus me looping dari tahap update ke draw sampai program ditutup karena setiap frame yang dilakukan oleh tahap update akan ditampilkan didalam draw.

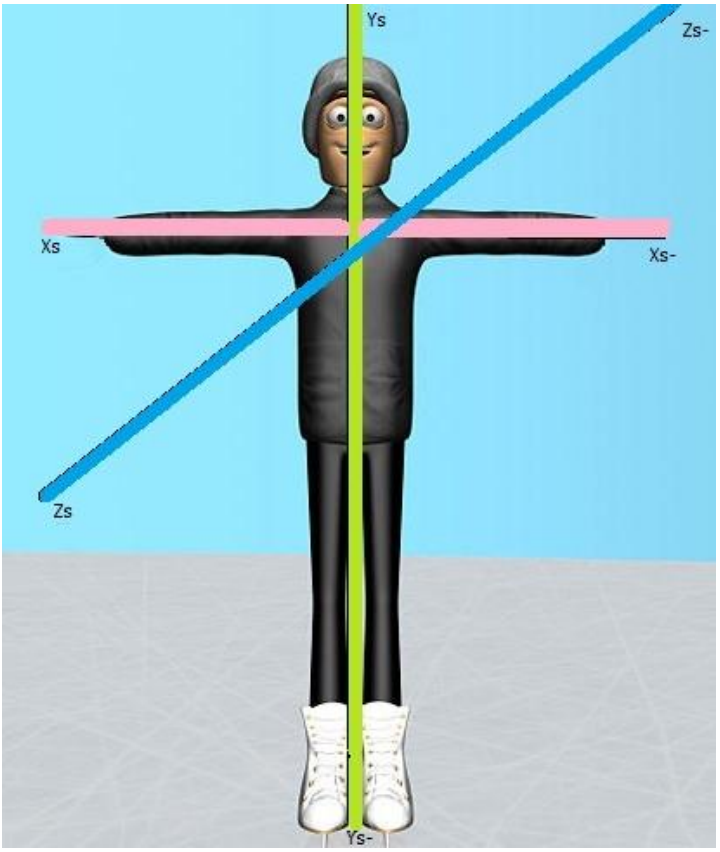
   Berikut tahapan yang dilakukan dalam draw:

   a. Persendian yang telah didefinisikan dinyatakan dalam koordinat Xs, Ys,dan Zs. Arah koordinat sumbu Xs positif dan negatif berada segaris ketika operator merentangkan tangan. Sumbu Xs positif segaris dengan tangan kanan dan sumbu Xs negatif segaris dengan tangan kiri. Sedangkan arah koordinat sumbu Ys positif berada di atas operator dan sumbu Ys negatif berada di bawah operator. Untuk koordinat sumbu Zs berada segaris antara operator berdiri dengan sensor Kinect, dengan sumbu positif berada di belakang dan negatif

didepan. Lebih jelasnya bisa dilihat pada Gambar 3.12 sampai dengan gambar 3.14.

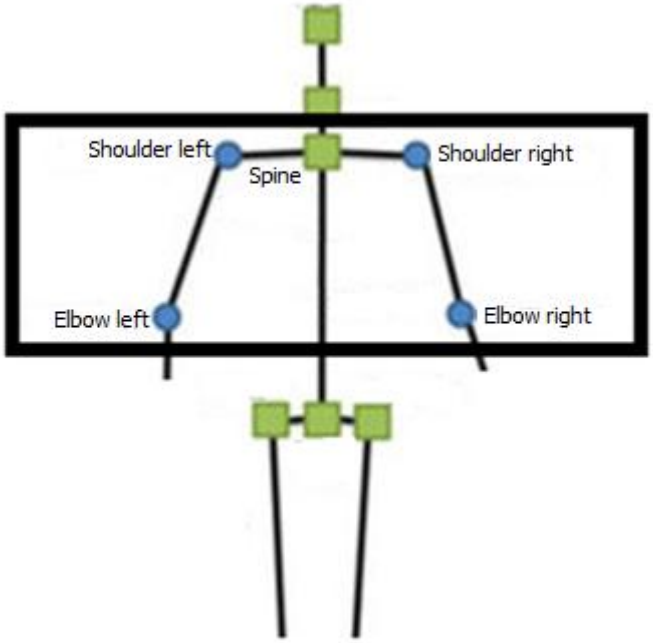


Gambar 3.12 arah kordinat ruang pada kinect



Gambar 3.13 Arah koordinat x, y dan z pada kinect

Berikut merupakan persendian yang digunakan didalam sistem:



Gambar 3.14 Persendian yang digunakan didalam sistem

3.1.6. Analisis Visualisasi

Pada analisis visualisasi, terdapat dua jenis bentuk tubuh yang akan digunakan. Yaitu bentuk tubuh Body Building dan bentuk tubuh Fitness model.

1. Bentuk tubuh Body building, disini kita menggunakan 5 gambar yang akan ditampilkan dalam proses visualisasi Body building, masing – masing gambar menampilkan bentuk tubuh yang berbeda beda sesuai dengan yang dipilih oleh member, kemudian juga dengan pola diet dan saran pelatihanya, setiap gambar yang dipilih untuk divisualisasikan dalam menu Body building akan menampilkan 1 menu pola diet dan 3 saran pelatihanya yang harus dilakukan setiap kali latihan.

   Penjelasan tentang pola diet, pelatihan yang disarankan dan jenis Body building dapat dilihat pada tabel 3.1

Tabel 3.1 Body building

No	Body Building	Pola diet	Jenis pelatihan
1		High frequent eating diet	1. Front dumbell raise 2. Front barbell raise 3. Overhead dumbell press
2		High frequent eating diet	1. Overhead dumbell press 2. Overhead barbell with smith machine 3. Bent over lateral raise
3		High frequent eating diet	1. Incline barbell press with smith machine 2. Seated bent over row 3. Front dumbell raise
4		High frequent eating diet	1. Flat barbell press 2. One arm dumbell row 3. Overhead dumbell press
5		High frequent eating diet	1. Front dumbell raise 2. Lat pul down 3. One arm dumbell row

2. Bentuk tubuh Fitness model, disini kita menggunakan 5 gambar yang akan ditampilkan dalam proses visualisasi, masing – masing gambar menampilkan bentuk tubuh yang berbeda beda sesuai dengan yang dipilih oleh member, akan tetapi pelatihan Fitness model tidak terlalu menekankan kepada latihan repetisi seperti pelatihan Body building. kemudian juga dengan pola diet dan saran pelatihanya, setiap gambar yang dipilih untuk divisualisasikan dalam menu Fitness model akan menampilkan 1 menu pola diet dan 3 saran pelatihanya yang harus dilakukan setiap kali latihan.

   Penjelasan tentang pola diet, pelatihan yang disarankan dan jenis Fitness model dapat dilihat pada tabel 3.2

Tabel 3.2 Fitness model

No	Fitness model	Pola diet	Jenis pelatihan
1		Atkins diet	1. Incline dumbell press 2. Pec dec 3. Cable crossover
2		Atkins diet	1. Pec dec 2. Incline barbell press with smith machine 3. Dumbel shrug
3		Atkins diet	1. Decline dumbell press 2. Decline barbell press 3. Overhead dumbell press

4		Atkins diet	1. One arm dumbell row 2. Overhead barbell press with smith machine 3. Front barbell raise
5		Intermitent diet fasting	1. Bent over lateral raise 2. Overhead barbell press with smith machine 3. Decline dumbell press

3.1.7. Analisis Jaringan

Pada analisis jaringan, jenis media yang akan digunakan berupa kabel UTP (Unshielded twisted pair) category 4 bertipe cross dengan menggunakan model jaringan peer to peer untuk menghubungkan komputer client dan komputer server, database member akan disimpan di komputer server dan dapat diakses oleh komputer client, ip yang digunakan adalah ip class c, ip komputer server ialah 192.168.1.1 dan ip yang komputer client ialah 192.168.1.2. analisis jaringan dapat dilihat pada Gambar 3.15



Gambar 3.15 Analisis jaringan

3.1.8. Perspektif produk

Aplikasi hasil dari pengolahan tubuh ini dibuat sebagai media yang ditujukan untuk membantu member yang ada di Rai fitness Purwakarta, selain itu juga sebagai media pelengkap dalam memvisualisasikan hasil dari pengolahan bentuk tubuhnya, dan tahapan-tahapannya.

1. Sistem yang digunakan harus terinstall software Kinect Sdk sebagai driver untuk mengkoneksikan sensor kinect minimal versi 1.8 dan Microsoft XNA versi 4.0 yang dapat di download langsung di situs resmi Microsoft.
2. Sambungkan Kinect ke komputer dan jalankan aplikasi, Sensor kinect menyediakan 20 kerangka titik sendi yang bisa kita panggil, Sensor kinect akan mendeteksi gerakan setiap frame. 2 frame sebelumnya, dan 2 frame berikutnya titik koordinat ditentukan dengan menggunakan sensor kedalaman yang dipancarkan oleh kamera inframerah.
3. Kemudian sistem mencocokkan titik sendi yang mana saja yang akan kita panggil didalam Microsoft XNA untuk divisualisasikan setelah titik sendi yang kita panggil terdeteksi maka visualisasi akan terjadi, tetapi jika titik sendi tersebut tidak terdeteksi maka visualisasi tidak akan terjadi.

3.1.8.1. Fitur dan Fungsi Produk

Fitur dan fungsi produk dapat dilihat pada tabel 3.3.

Tabel 3.3 Fitur dan deskripsi fungsi aplikasi

	Fitur	Deskripsi Fungsi
1	Visualisasi	Fitur untuk menampilkan hasil dari visualisasi kemungkinan bentuk tubuh.
		Fitur untuk untuk menampilkan pola diet dan saran pelaihan berdasarkan jenis bentuk tubuh yang dipilih.
		Data visualisasi berisi foto foto body building yang digunakan dalam sistem.
2	Data Statistik	Fitur untuk mengecek perubahan berat badan dan tinggi badan serta foto diri yang disimpan pada setiap kali latihan.
		Fitur untuk menampilkan data statistik member yang disimpan di komputer server.

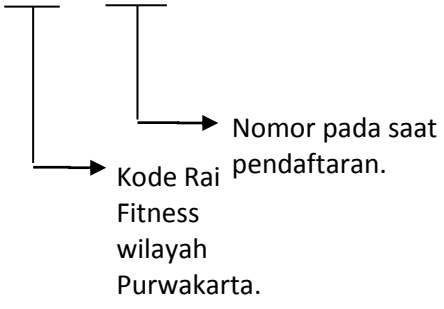
3.1.9. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang menitikberatkan pada properti yang digunakan oleh sistem, diantaranya kebutuhan perangkat keras, perangkat lunak, serta user sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.1.9.1. Analisis Pengkodean

Pada proses pengkodean aplikasi pengolahan tubuh ini, terdapat satu pengkodean yang diusulkan yaitu pengkodean produk seperti berikut :

Format Id Member : 64 - 000



Contoh : 64-001 , merupakan Nomor anggota Rai Fitness wilayah Purwakarta dengan nomor pendaftaran pertama

3.1.9.2. Analisis Perangkat Lunak

Perangkat lunak digunakan dalam sebuah sistem yang merupakan sebuah himpunan perangkat yang mendukung atau dibutuhkan untuk membangun aplikasi ini dapat dilihat pada tabel 3.4 dan tabel 3.5:

Tabel 3.4 Fitur dan deskripsi fungsi perangkat lunak sisi client

	Fitur	Deskripsi Fungsi
1.	Sistem operasi windows 7	Fitur untuk menjalankan program yang telah dibangun
2.	Kinect SDK 1.8	Toolkit yang digunakan dalam membangun program.
3.	*Corel draw x3*	Aplikasi pendukung yang digunakan untuk membuat program.
4.	Adobe Photoshop Cs3	Aplikasi Pendukung yang digunakan untuk membuat program.
5.	Google sketchup 5.0	Aplikasi Pendukung yang digunakan untuk menyusun laporan.
6.	Microsoft Visual Studio 2010	Bahasa pemrograman yang digunakan dalam membangun program.
7.	Microsoft Xna Framework 4.0	Toolkit yang digunakan dalam membangun program

Tabel 3.5 fitur dan deskripsi perangkat lunak sisi server

	Fitur	Deskripsi Fungsi
1.	Sistem operasi windows 7	Fitur untuk menjalankan program yang telah dibangun
2.	Microsoft access 2010	Aplikasi yang digunakan untuk membuat database
3.	Adobe Photoshop Cs3	Aplikasi Pendukung yang digunakan untuk mendesain gambar.

3.1.9.3. Analisis Perangkat Keras

Komputer dan Kinect adalah sebuah perangkat keras yang tidak luput dari perangkat lunak sebagai interaksinya. Perangkat lunak memberikan sebuah perintah-perintah terhadap perangkat keras agar dapat berjalan dengan baik. Dalam pembangunan aplikasi ini menggunakan perangkat keras sebagai pendukungnya dapat dilihat pada tabel 3.6 dan tabel 3.7:

Tabel 3.6 Fitur dan deskripsi fungsi perangkat keras sisi client

	Fitur	Deskripsi Fungsi
1.	Processor Pentium dualcore 3.0 GHZ	Spesifikasi minimal processor yang dibutuhkan untuk menjalankan program.
2.	Ram 4 GB	Spesifikasi minimal ram yang dibutuhkan dalam menjalankan program.
3.	Mouse dan Keyboard	Kontrol minimal yang dibutuhkan dalam sistem.
4.	Kinect Xbox 360	Perangkat pendukung yang dibutuhkan dalam menjalankan program.

Tabel 3.7 Fitur dan deskripsi fungsi perangkat keras sisi server

	Fitur	Deskripsi Fungsi
1.	Processor Pentium dualcore 2.0 GHZ	Spesifikasi minimal processor yang dibutuhkan untuk menjalankan program.
2.	Ram 1 GB	Spesifikasi minimal ram yang dibutuhkan dalam menjalankan program.
3.	Mouse dan Keyboard	Kontrol minimal yang dibutuhkan dalam sistem.

3.1.10. Analisis Kebutuhan Fungsional

Kebutuhan fungsional berhubungan dengan fitur software yang akan dibuat atau dikembangkan. Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan agar sistem dapat berjalan dengan baik.

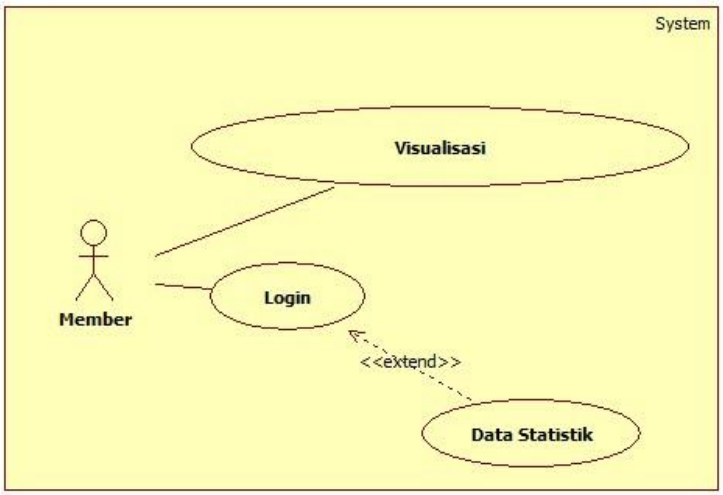
Analisis yang dilakukan dimodelkan dengan menggunakan UML (Unified Modeling Language). Tahap-tahap pemodelan dalam analisis tersebut antara lain identifikasi aktor, *usecase* diagram, skenario, *sequence* diagram, *activity* diagram, *class* diagram, dll.

3.1.10.1. Use case diagram

Pada sub bab berikut ini dijelaskan bagaimana *Use Case* yang terjadi dalam pembangunan aplikasi ini yang dimodelkan dalam sekumpulan *Use Case, Actor* dan bagaimana hubungan - hubungannya yang akan dipetakan dalam diagram *Use Case*, dilengkapi dengan skenario untuk menjelaskan dari gambaran *Use Case* yang ada dengan menggambarkan skenario per *Use Case*. *Use Case Diagram* dapat dilihat pada Gambar 3.16 dan Gambar 3.17

1. Use case pada sisi client.

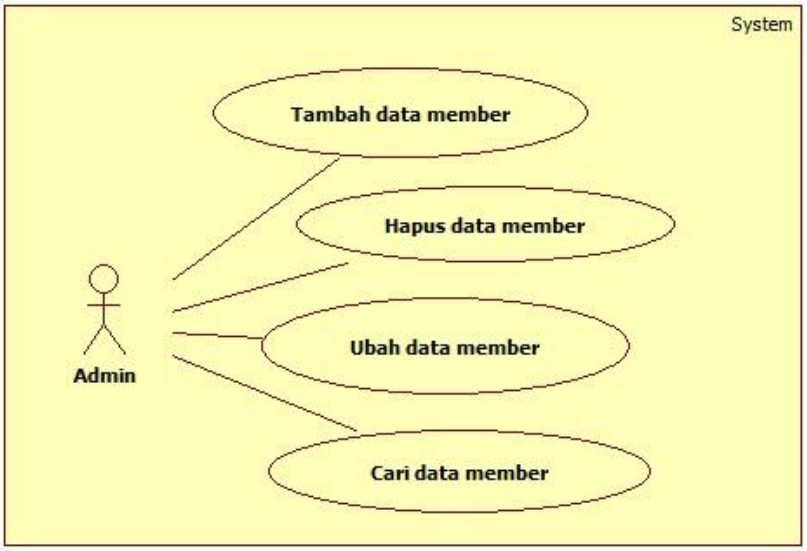
   Use case aplikasi di komputer client dijelaskan pada gambar 3.16



Gambar 3.16 Use case client

2. Use case pada sisi server.

Use case diagram pada sisi server dapat dilihat pada gambar 3.17



Gambar 3.17 Use case server

1. Actor Definiton

Actor Definiton berfungsi untuk menjelaskan *actor* yang terdapat pada *Use Case Diagram* dan menggambarkan pihak-pihak yang berperan dalam sistem.

2. Use Case Definition

Use Case definition berfungsi untuk menjelaskan fungsi *Use Case* yang terdapat pada *Use Case Diagram* dan menjelaskan aktivitas yang disiapkan oleh sistem. *Use Case Definition* diterangkan pada Tabel 3.8 dan tabel 3.9

Tabel 3.8 Use case definition sisi client

No	Use Case	Deskripsi
1.	Visualisasi	Proses memvisualisasikan Kemungkinan bentuk tubuh.
2.	Login	Proses untuk memasukan nomor anggota
3.	Data Statistik	Proses untuk melihat perubahan bentuk tubuhnya dari foto diri yang disimpan.

Tabel 3.9 Use case definition sisi server

No	Use Case	Deskripsi
1.	Tambah data member	Proses untuk menambah data member
2	Hapus data member	Proses untuk menghapus data member
3	Ubah data member	Proses untuk mengubah atau memodifikasi data member
4	Cari data member	Proses untuk mencari data member

3. Use Case Scenario

Use Case Scenario mendeskripsikan urutan langkah-langkah dalam proses bisnis baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor.

1. Use Case scenario sisi client dijelaskan dari point a sampai dengan point c.

a. Use Case Scenario Visualisasi

Use case scenario visualisasi bentuk tubuh dijelaskan oleh tabel 3.10

Tabel 3.10 Use case scenario visualisasi bentuk tubuh

Identifikasi	
Nomor	1
Nama	Visualisasi
Tujuan	Proses yang dilakukan user untuk memvisualisasikan kemungkinan bentuk tubuh.
Aktor	Member
Skenario utama	
Kondisi awal	Member berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu visualisasi.	2.Membuka menu visualisasi. 3.Mendeteksi kinect yang terhubung. 4.Kinect telah terdeteksi.
5. Memilih jenis bentuk tubuh.	6.Sistem mendeteksi tubuh 7.Sistem memvisualisasikan bentuk tubuh. 8.Sistem menampilkan gambar visualisasi yang dipilih. 9.Sistem menampilkan pola diet dan saran pelatihan.
Skenario Alternatif	
Kondisi Aktor	**Reaksi system**
1.Memilih menu visualisasi bentuk tubuh.	2.Membuka menu visualisasi bentuk tubuh. 3.Mendeteksi kinect yang terhubung. 4.Kinect tidak terdeteksi.

	5.Menampilkan Sambungkan kinect.
Kondisi akhir	Masa otot berhasil tervisualisasi

b. **Use Case Scenario Login**

Use case scenario login dijelaskan oleh tabel 3.11

Tabel 3.11 Use case scenario login

Identifikasi	
Nomor	2
Nama	Login
Tujuan	Proses yang dilakukan member untuk memasukan id member.
Aktor	Member
Skenario utama	
Kondisi awal	Member berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu login	2.Membuka menu login.
3.Memasukan id member berdasarkan kartu anggota.	4.Sistem mengecek ada tidaknya data. 5.Sistem menemukan data member. 6.Menampilkan Selamat datang.
Skenario Alternatif - Autentikasi gagal	
Kondisi Aktor	**Reaksi Sistem**
1.Memilih menu login	2.Membuka menu login.
3.Memasukan id member berdasarkan kartu anggota.	4.Sistem mengecek ada tidaknya data. 5.Sistem tidak menemukan data member. 6.Menampilkan data tidak tersedia.
Kondisi akhir	Login telah berhasil.

c. Use Case Scenario Data statistic

Use case scenario data statistik dijelaskan oleh tabel 3.12

Tabel 3.12 Use case scenario data statistik

Identifikasi	
Nomor	3
Nama	Data statistik
Tujuan	Melihat data statistik member.
Aktor	Member
Skenario utama	
Kondisi awal	Member telah berhasil login
Kondisi Aktor	**Reaksi system**
1.Memilih menu data statistik	
	2.Sistem mencari data statistik member di database. 3.Sistem menampilkan data statistik member dan foto yang disimpan di database. 4.Menampilkan tombol ambil gambar
5.Member menekan tombol Ambil gambar.	6.Sistem mendeteksi kamera dan mengambil foto terbaru member.
Kondisi akhir	Menampilkan data statistik member dan foto diri member yang disimpan di database.

2. Use case scenario sisi server dijelaskan melalui poin d sampai dengan g.

d. Use Case Scenario Tambah data member

Use case scenario Tambah data member dijelaskan oleh tabel 3.13

Tabel 3.13 use case scenario Menambah data member

Identifikasi	
Nomor	4
Nama	Tambah data member
Tujuan	Menginputkan data member baru kedalam system
Aktor	Admin
Skenario utama	
Kondisi awal	Admin berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu input data member.	2.sistem membuka menu input data member
3.Menginputkan data member.	4.Sistem mengecek apakah ada data member yang sama. 5.Sistem tidak menemukan data yang sama. 6.Menampilkan Data member telah disimpan
Skenario Alternatif	
Kondisi Aktor	**Reaksi system**
1.Memilih menu input data member.	2.sistem membuka menu input data member

3.Menginputkan data member.	4.Sistem mengecek apakah ada data member yang sama. 5.Sistem menemukan data yang sama. 6.Menampilkan data member yang sama sudah ada.
Kondisi akhir	Data member telah berhasil disimpan.

e. Use Case Scenario Hapus data member

Use case scenario Hapus data member dijelaskan oleh tabel 3.14

Tabel 3.14 Use case scenario mengapus data member

Identifikasi	
Nomor	5
Nama	Menghapus data member
Tujuan	Menghapus data member baru kedalam system
Aktor	Admin
Skenario utama	
Kondisi awal	Admin berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu hapus data member	2.Membuka menu hapus data member
3.Memasukan id member	4.Sistem mengecek ada atau tidaknya data member. 5.Sistem menemukan data member. 6.Sistem menampilkan tombol hapus.
7.Memilih tombol hapus	
	8.Menampilkan tombol yakin dan batal
9.Memilih tombol yakin	10.Menghapus data member dari

	system. 11.Menampilkan Data member talah dihapus
Skenario alternatif	
Kondisi Aktor	**Reaksi system**
1.Memilih menu hapus data member	2.Membuka menu hapus data member
3.Memasukan id member	4.Sistem mengecek ada atau tidaknya data member. 5.Sistem tidak menemukan data member. 6.Sistem menampilkan data member telah dihapus..

f. Use Case Scenario Ubah data member

Use case scenario Ubah data member dijelaskan oleh tabel 3.15

Tabel 3.15 Use case scenario merubah data member

Identifikasi	
Nomor	6
Nama	Mengubah data member
Tujuan	Mengubah data member yang ada didalam system
Aktor	Admin
Skenario utama	
Kondisi awal	Admin berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu Ubah data member	2.Membuka menu ubah data member
3.Memasukan id member	4.Sistem mengecek ada atau tidaknya

	data member. 5.Sistem menemukan data member. 6.Sistem menampilkan tombol edit.
7.Memilih tombol edit	8.Membuka data member 9.Sistem menampilkan menu simpan dan batal.
10.Merubah data yang telah terisi. 11.Memilih tombol simpan.	12.Menyimpan data ke system. 13.Menampilkan data member telah diperbahrui.
Skenario alternatif	
Kondisi Aktor	**Reaksi system**
1.Memilih menu Ubah data member	2.Membuka menu ubah data member
3.Memasukan id member	4.Sistem mengecek ada atau tidaknya data member. 5.Sistem tidak menemukan data member. 6.Menampilkan data member tidak ditemukan.
Kondisi akhir	Data member telah berhasil diubah.

g. Use case scenario Cari data member

Use case scenario Cari data member dijelaskan oleh tabel 3.16

Tabel 3.16 Use case scenario cari data member

Identifikasi	
Nomor	7
Nama	Cari data member
Tujuan	Mencari data member yang ada didalam system
Aktor	Admin
Skenario utama	
Kondisi awal	Admin berada di menu utama
Kondisi Aktor	**Reaksi system**
1.Memilih menu Cari data member	2.Membuka menu Cari data member
3.Memilih jenis pencarian 4.Memasukan kata kunci	5.Sistem mengecek ada atau tidaknya data member. 6.Sistem menemukan data member. 7.Sistem menampilkan data member.
Skenario alternatif	
Kondisi Aktor	**Reaksi system**
1.Memilih menu Cari data member	2.Membuka menu Cari data member
3.Memilih jenis pencarian 4.Memasukan kata kunci	5.Sistem mengecek ada atau tidaknya data member. 6..Sistem tidak menemukan data member. 7.Menampilkan pesan data member tidak ditemukan.
Kondisi akhir	Data member telah ditemukan.

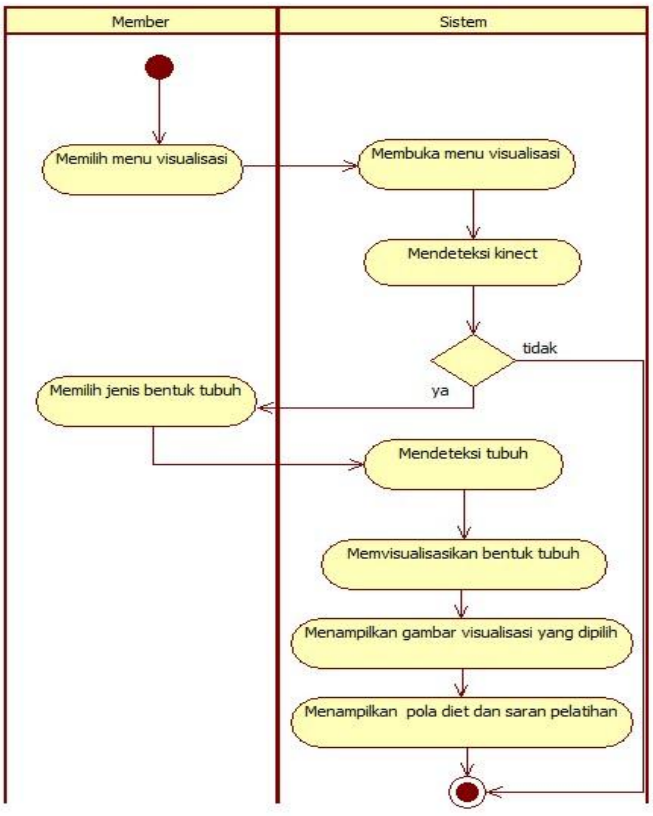
3.1.11. Activity Diagram

Activity Diagram lebih memfokuskan diri pada eksekusi dan alur sistem dari pada bagaimana sistem itu dirakit. Diagram ini tidak hanya memodelkan *software* melainkan memodelkan model bisnis juga. *Activity Diagram* menunjukan aktivitas sistem dalam bentuk kumpulan aksi-aksi. Berikut ini *Activity Diagram* pada Aplikasi hasil dari pengolahan tubuh.

a. Activity diagram sisi client dijelaskan melalui point 1 sampai dengan 3

1. Activity diagram visualisasi

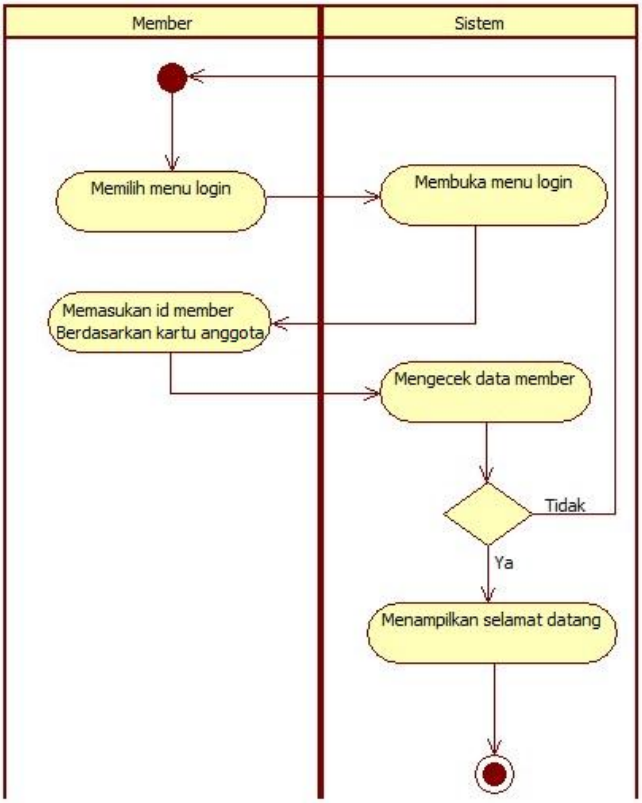
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan visualisasi bentuk tubuh yang dapat dilihat pada gambar 3.18.



Gambar 3.18 Activity diagram Visualisasi bentuk tubuh.

2. Activity diagram Login

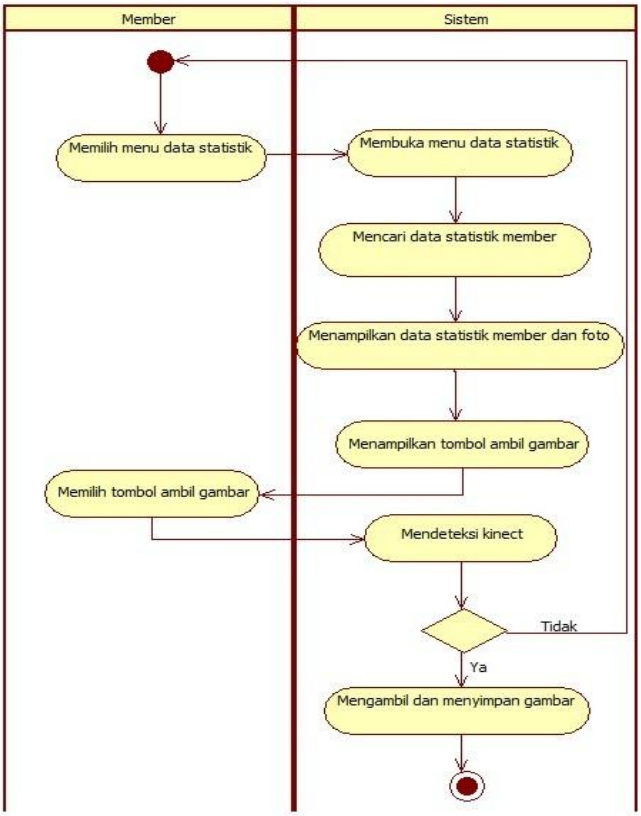
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Login yang dapat dilihat pada gambar 3.19.



Gambar 3.19 Activity diagram Login

3. Activity diagram Data statistik

Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Data statistik yang dapat dilihat pada gambar 3.20.

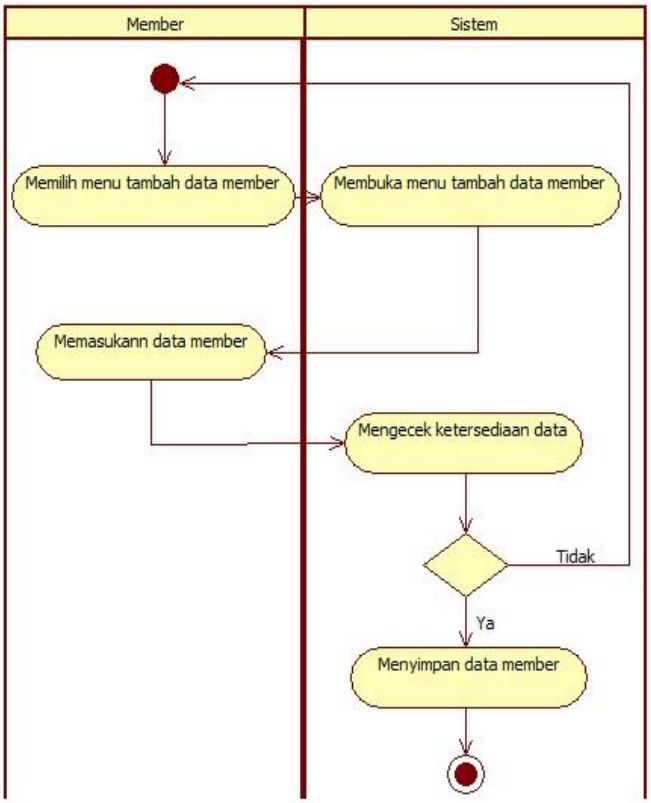


Gambar 3.20 Activity diagram data statistik

b. Activity diagram sisi server dijelaskan melalui point 4 sampai dengan 6

4.Activity diagram Tambah data member

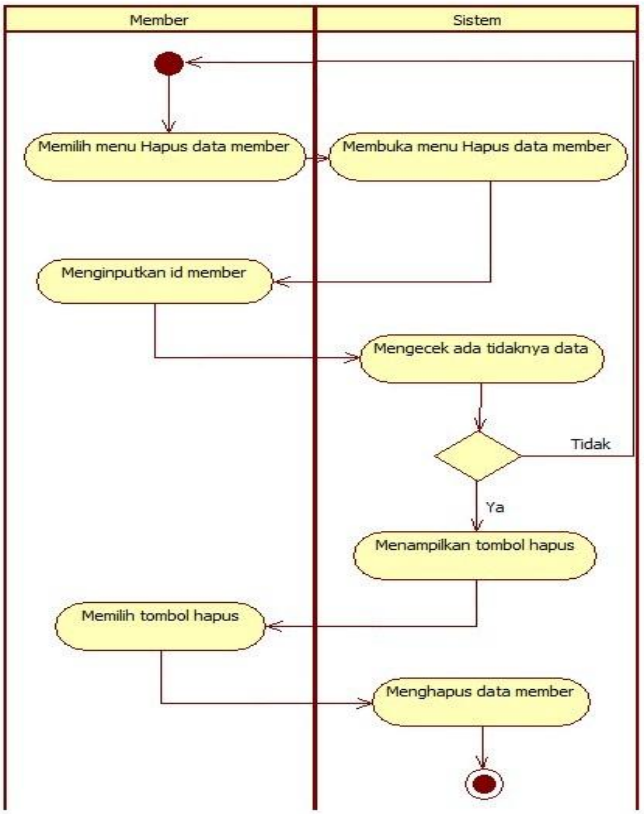
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Menambah data member yang dapat dilihat pada gambar 3.21.



Gambar 3.21 Activity diagram Menambah data member

5.Activity diagram Hapus data member

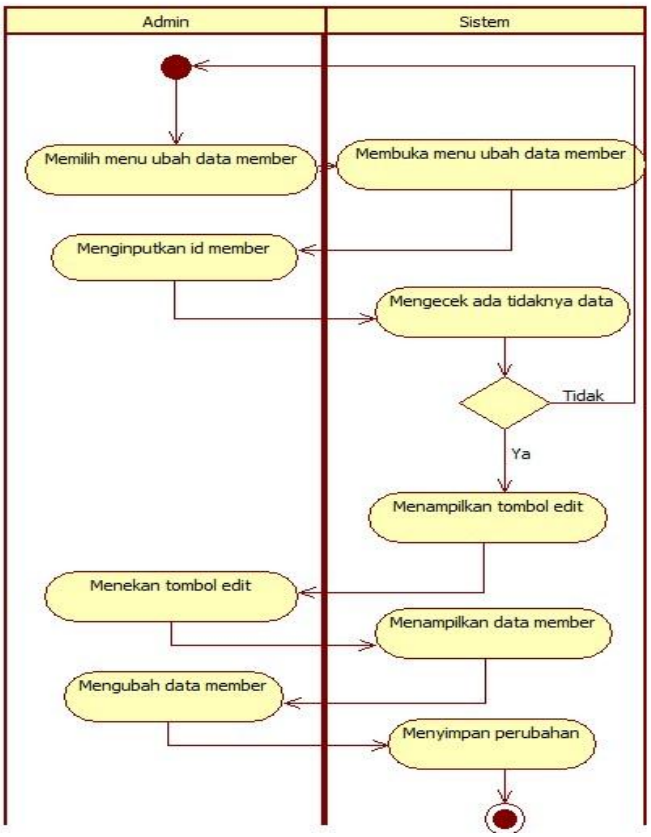
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Menghapus data member yang dapat dilihat pada gambar 3.22



Gambar 3.22 Activity diagram Menghapus data member

6.Activity diagram Ubah data member

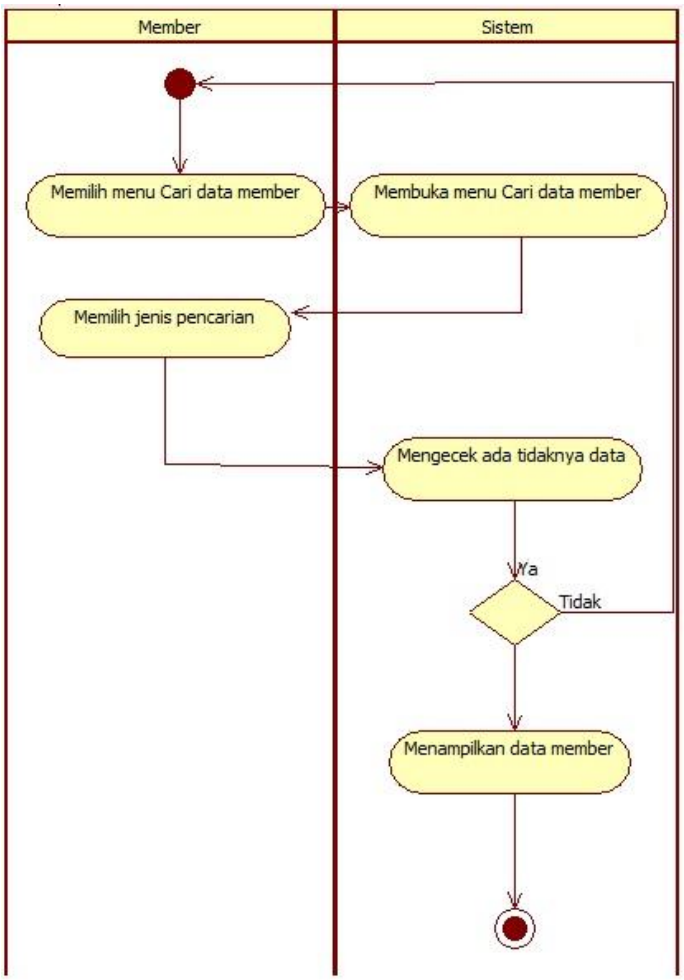
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Mengubah data member yang dapat dilihat pada gambar 3.23.



Gambar 3.23 Activity diagram Mengubah data member

7.Activity diagram Cari data member

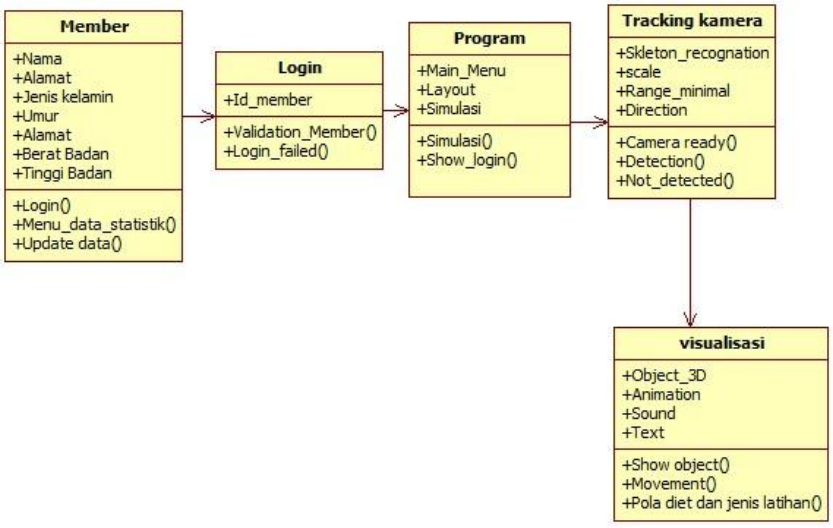
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas menampilkan Mengubah data member yang dapat dilihat pada gambar 3.24.



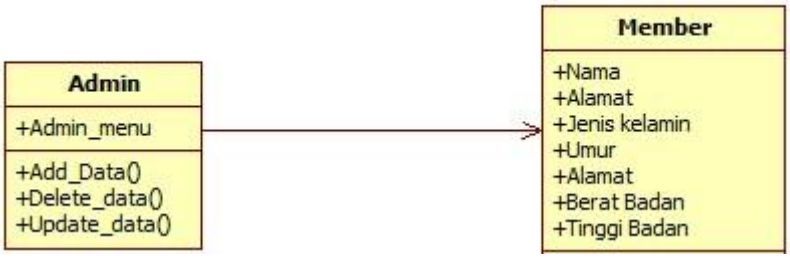
Gambar 3.24 Activity diagram cari data member

3.1.12. Class Diagram

Class diagram adalah inti dari proses pemodelan objek. Class Diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem. Struktur itu meliputi atribut dan method yang ada pada masing masing class. Class Diagram aplikasi hasil dari pengolahan tubuh dapat dilihat pada gambar 3.25 sampai dengan 3.26



Gambar 3.25 Class diagram sisi client



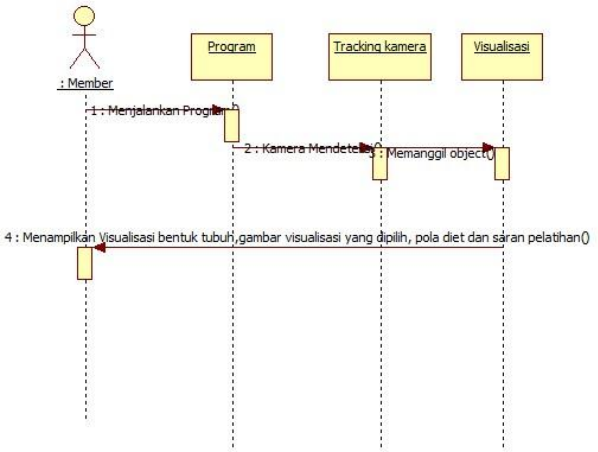
Gambar 3.26 Class diagram sisi server

3.1.13. Sequence Diagram

Sequence Diagram menggambarkan interaksi antar masing-masing objek pada setiap use case dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi. Sequence Diagram pada aplikasi hasil dari pengolahan tubuh terdiri dari sequence diagram memvisualisasikan masa otot, sequence diagram data statistik, sequence diagram login, sequence diagram menambah data member, sequence diagram menghapus data member, sequence diagram merubah dara member. Berikut penjelasan dari masing-masing sequence diagram dilihat dalam gambar 3.27 sampai dengan 3.33 :

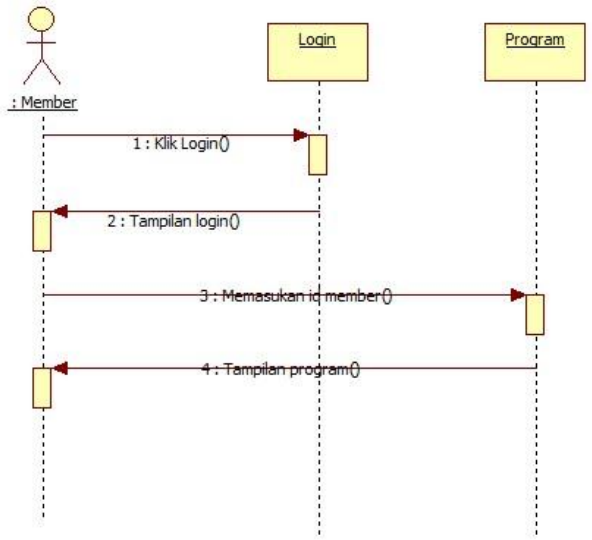
A. *Squence* diagram sisi client dijelaskan pada point 1 sampai dengan 3

1. *Squence diagram* menu visualisasi pada gambar gambar 3.27.



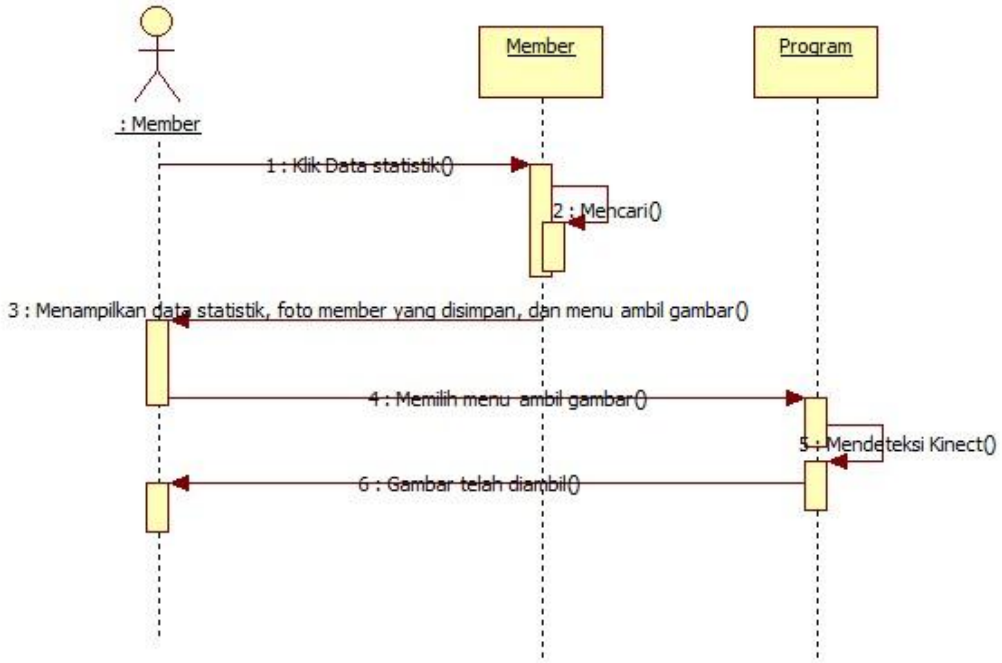
Gambar 3.27 Sequence diagram menu visualisasi

2. *Squence diagram* menu login dilihat pada gambar 3.28:



Gambar 3.28 Sequence diagram menu login

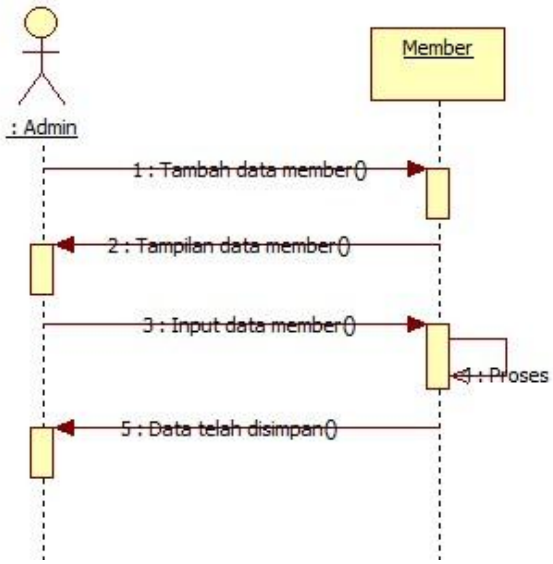
3. Squence diagram Data statistik dilihat pada gambar 3.29:



Gambar 3.29 Sequence diagram data statistik

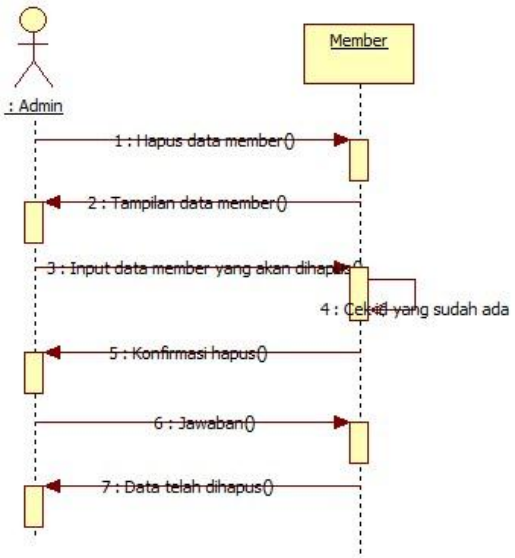
B. Squence diagram sisi server dijelaskan pada point 4 sampai dengan 6.

4. Squence diagram Tambah data member dilihat pada gambar 3.30



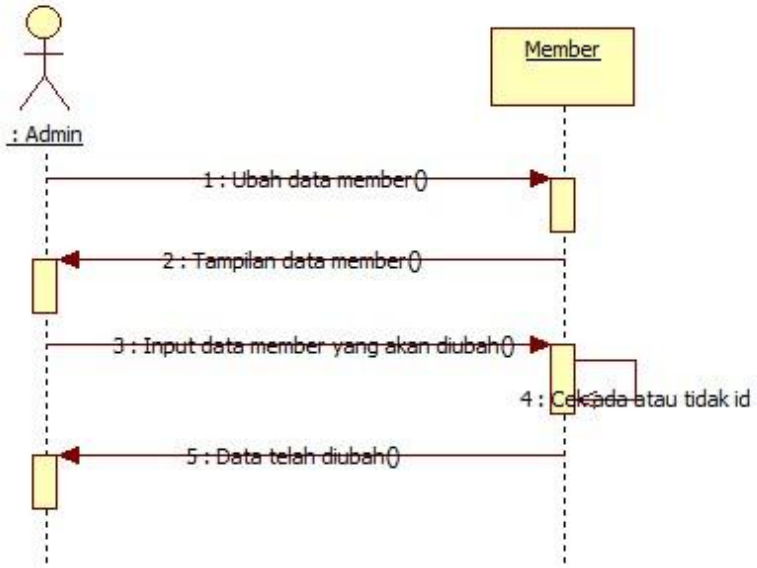
Gambar 3.30 Sequence diagram tambah data member

5. Squence diagram Hapus data member dilihat pada gambar 3.31



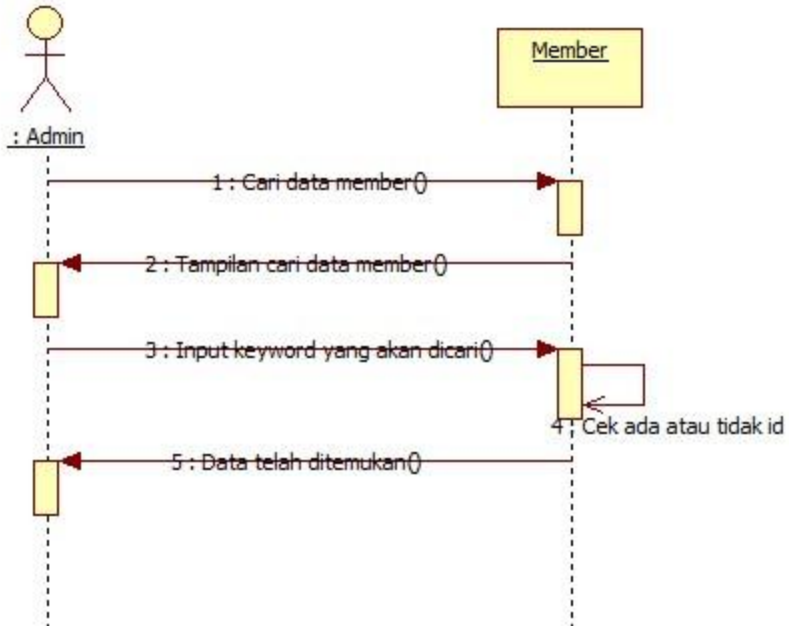
Gambar 3.31 Sequence diagram menghapus data member

6. Squence diagram Ubah data member dilihat pada gambar 3.32



Gambar 3.32 Sequence diagram mengubah data member

7. Squence diagram Cari data member dilihat pada gambar 3.33



Gambar 3.33 Sequence diagram mencari data member

3.2. Perancangan Sistem

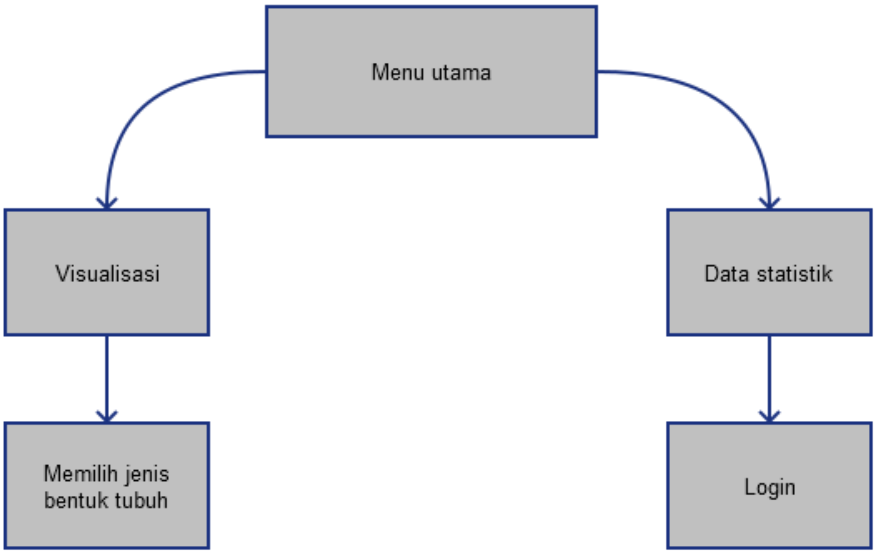
Perancangan sistem yang akan dilakukan meliputi tiga tahap yaitu:

1. Perancangan Struktur menu
2. Perancangan Struktur tabel
3. Perancangan antarmuka (interface)
4. Perancangan pesan

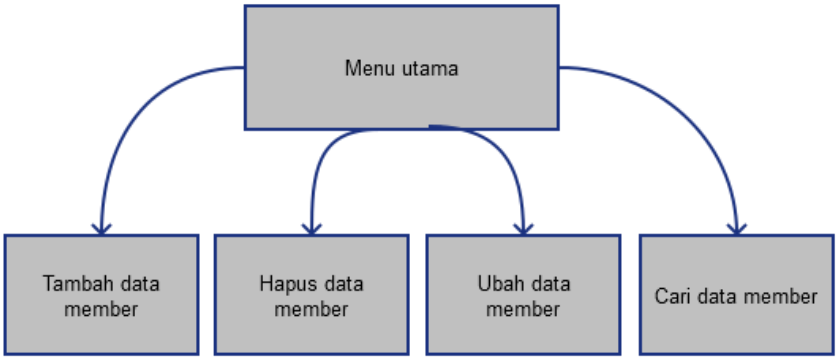
Berikut penjelasan mengenai perancangan sistem:

3.2.1 Perancangan Struktur Menu

Untuk mempermudah dalam pembuatan sistem, maka diperlukan perancangan struktur menu program yang akan dibangun. Perancangan struktur menu program ini membantu dalam merancang bagian-bagian dari sistem yang sebenarnya dan untuk mengetahui bagian mana yang nantinya akan diakses terlebih dahulu setelah program selesai dibuat. Perancangan struktur menu dapat dilihat pada gambar 3.34 sampai dengan 3.35.



Gambar 3.34 Struktur menu sisi client



Gambar 3.35 Struktur menu sisi server

3.2.2. Perancangan Struktur tabel

Tabel yang terdapat di sistem ini adalah hanya dua tabel yaitu tabel data member dan tabel statistic perancangan struktur tabel dapat dilihat pada tabel 3.17 dan tabel 3.18

Tabel 3.17 Struktur tabel data member

Nama *field*	Tipe data	Ukuran	Keterangan
Member_id	Varchar	100	Primary Key
Name	Varchar	100	
Gender	Varchar	20	
Age	Int	4	
Address	Varchar	100	
Height	Int	4	
Weight	Int	4	

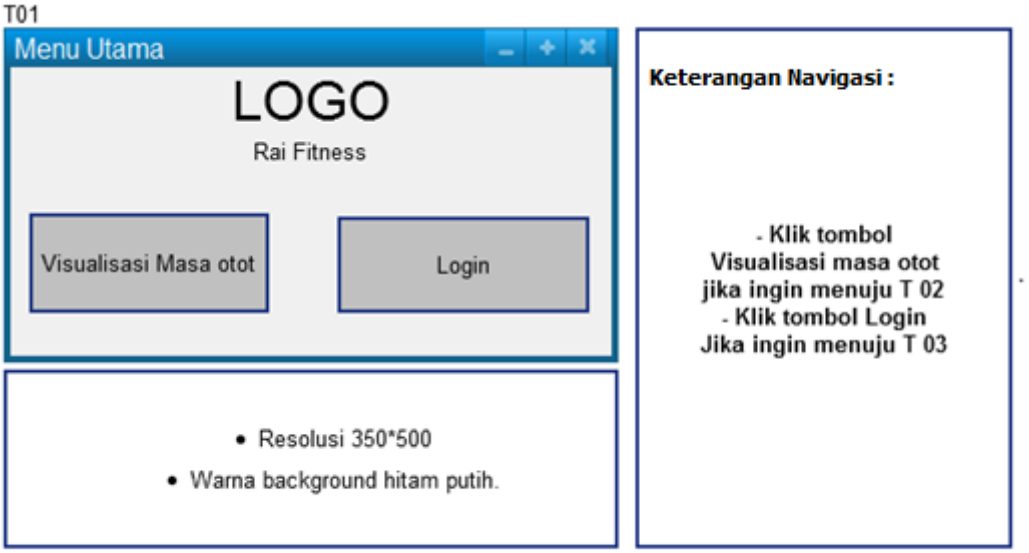
Tabel 3.18 Struktur tabel data statistic

Nama *field*	Tipe data	Ukuran	Keterangan
Height	Int	4	
Weight	Int	4	

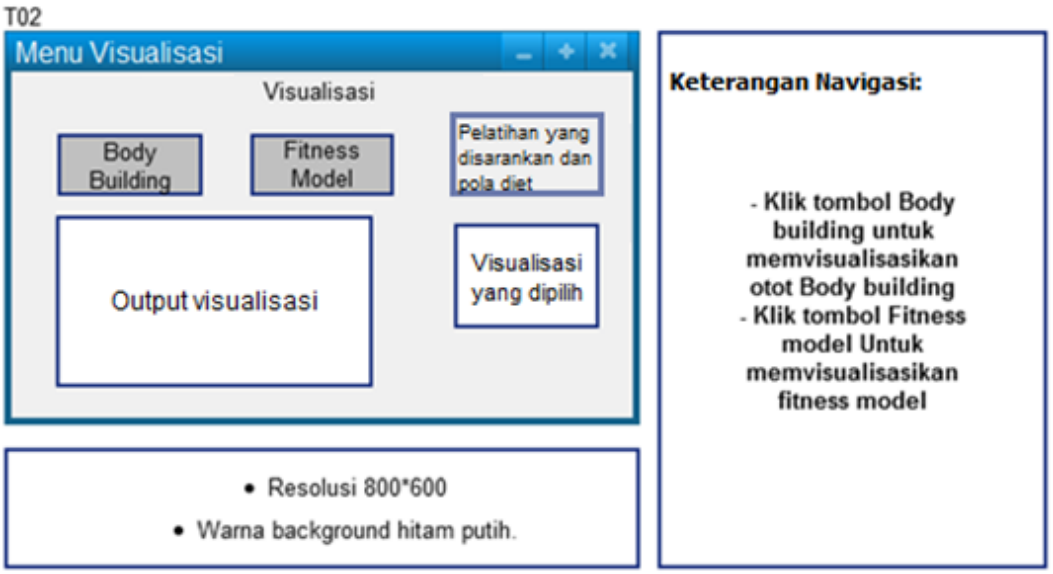
3.2.3. Perancangan Antar muka

Perancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan mempermudah dalam mengimplementasikan aplikasi serta akan memudahkan pembuatan aplikasi. Antarmuka aplikasi merupakan halaman yang muncul pertama kali pada saat menjalankan aplikasi. Perancangan antarmuka dijelaskan pada gambar 3.36 – 3.45 :

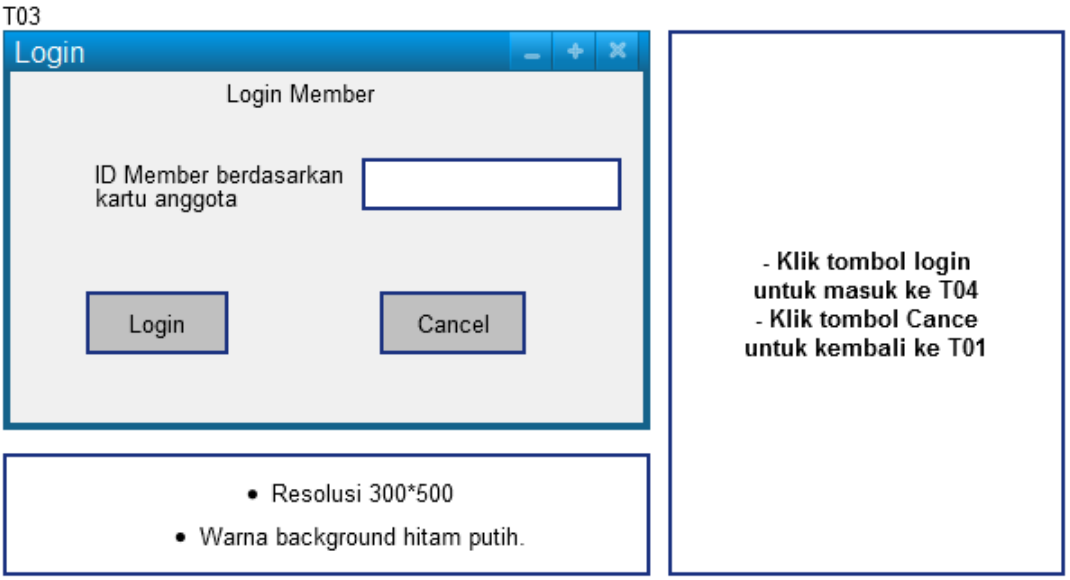
1. Perancangan antarmuka pada Sisi Client dijelaskan oleh gambar 3.36 sampai dengan gambar 3.38



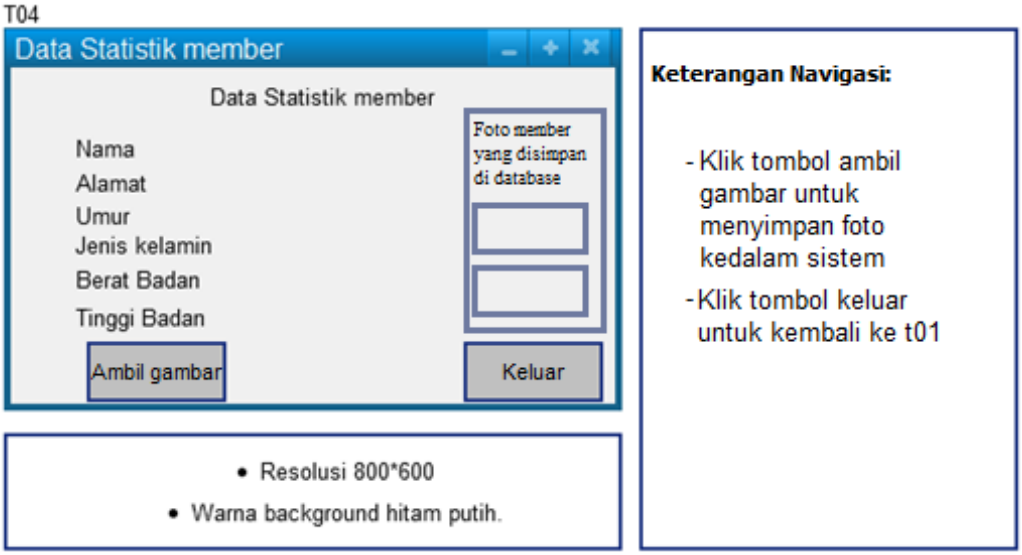
Gambar 3.36 Tampilan menu utama



Gambar 3.37 Tampilan antarmuka menu visualisasi

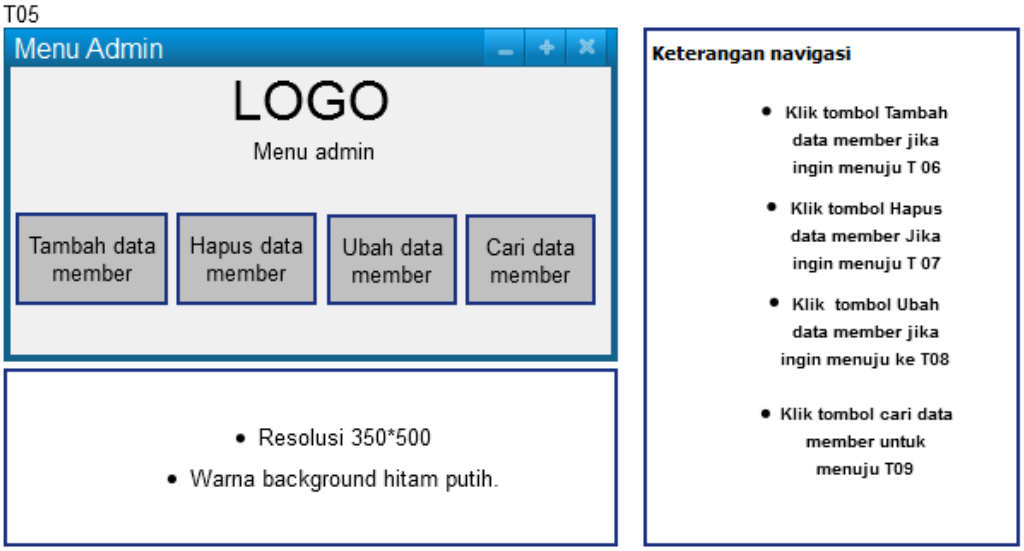


Gambar 3.38 Tampilan antarmuka login

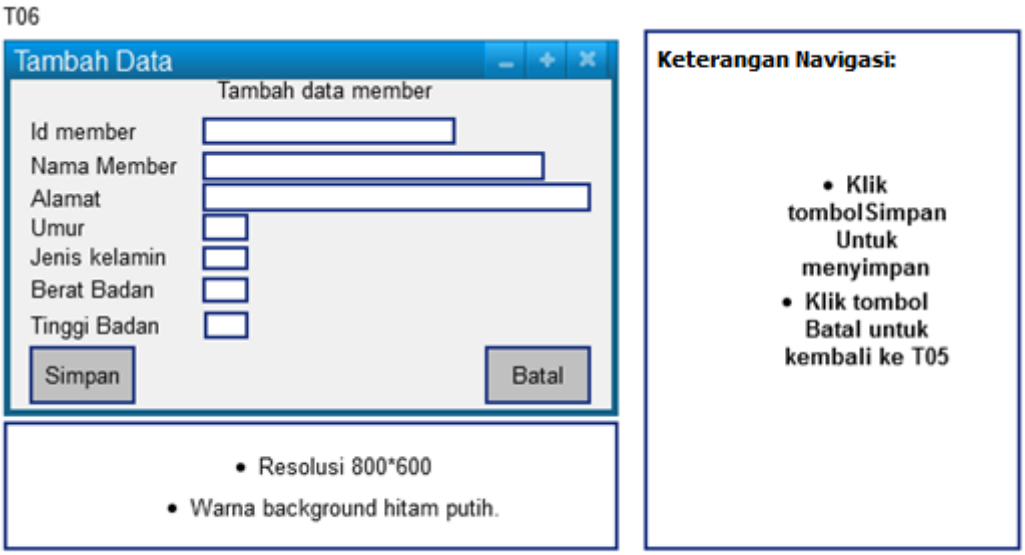


Gambar 3.39 Tampilan antarmuka data statistik

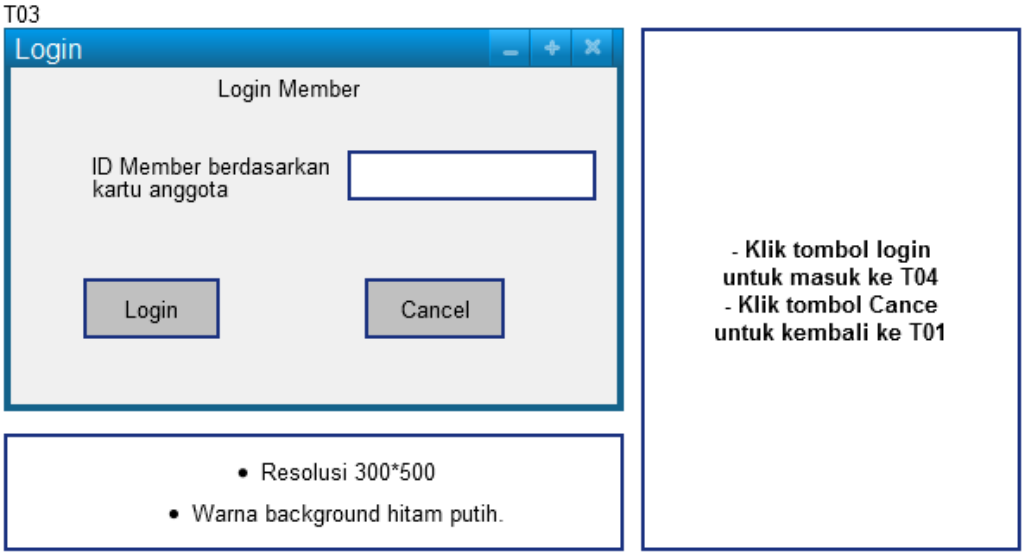
2. Perancangan antarmuka pada Sisi server dijelaskan oleh gambar 3.40 sampai dengan gambar 3.44



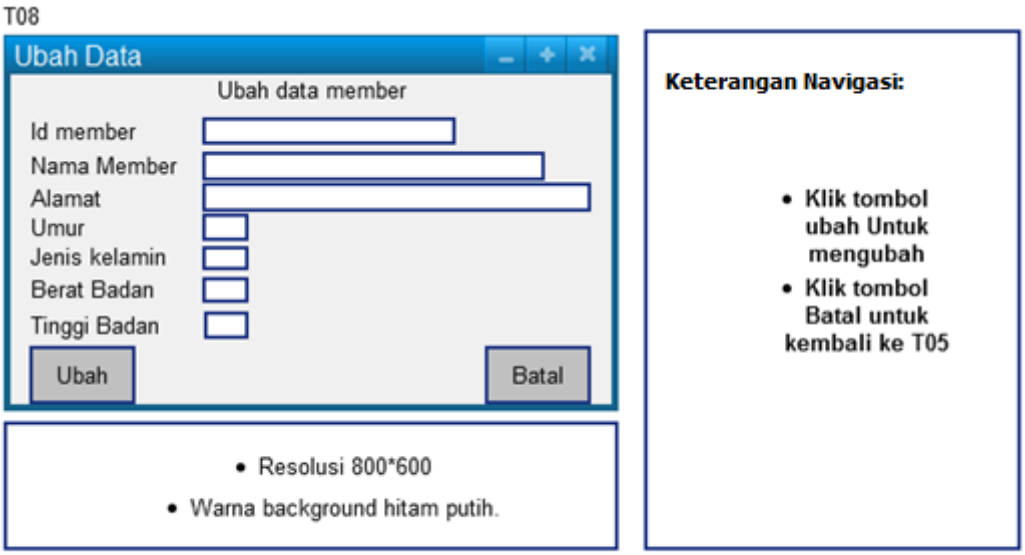
Gambar 3.40 Tampilan antarmuka menu admin



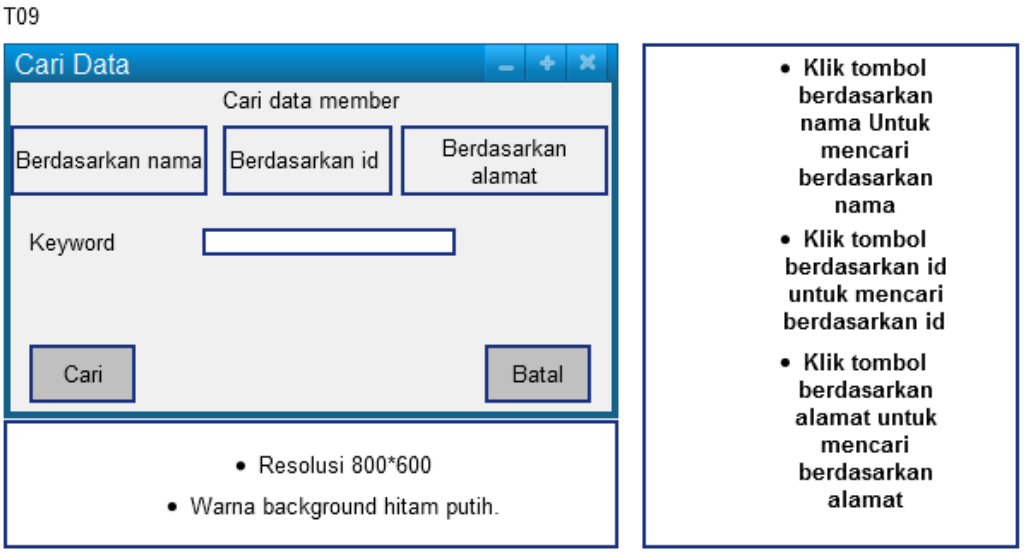
Gambar 3.41 Tampilan antarmuka tambah data



Gambar 3.42 Tampilan antarmuka hapus data



Gambar 3.43 Tampilan antarmuka ubah data



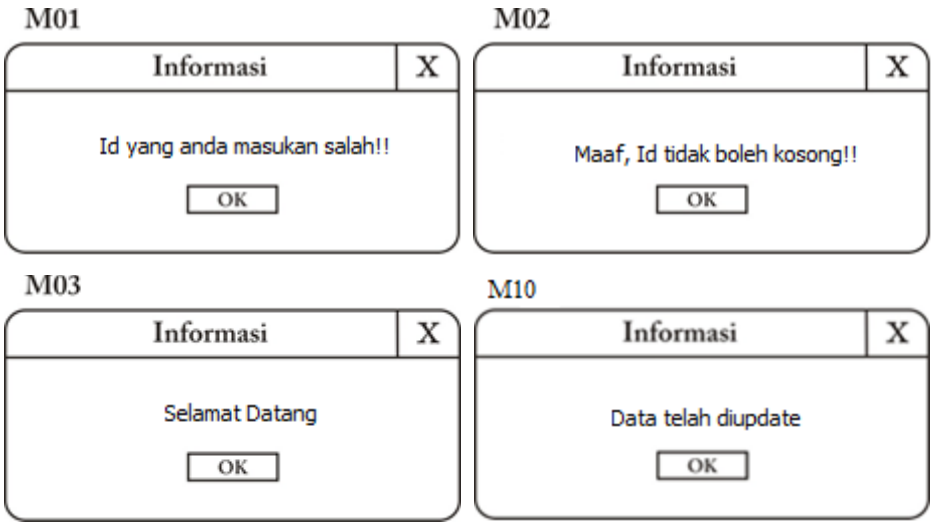
Gambar 3.44 Tampilan antarmuka cari data

3.2.4. Perancangan pesan

Perancangan antarmuka pesan pada Aplikasi interaktif hasil dari pengolahan tubuh ini dapat dilihat pada gambar 3.45 sampai dengan gambar 3.46 :

A.Sisi client

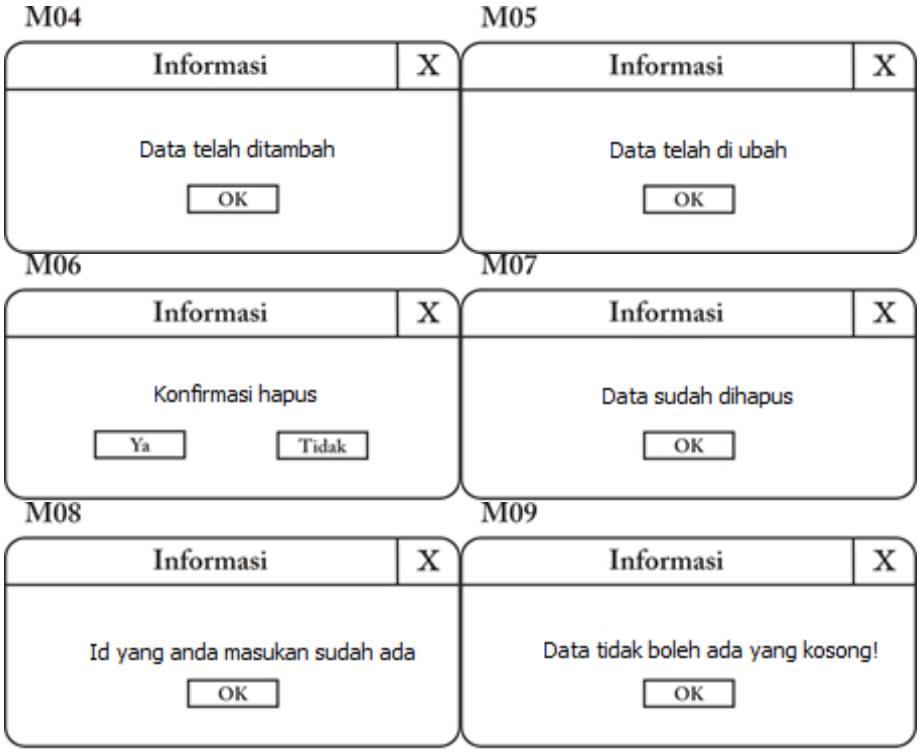
Berikut perancangan antar muka pesan sisi client



Gambar 3.45 Rancangan antarmuka pesan sisi client

B.Sisi server

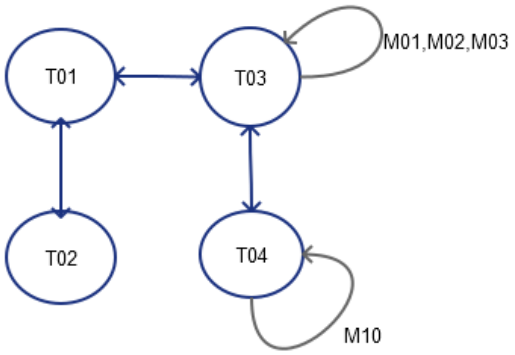
Berikut perancangan antarmuka pesan sisi server.



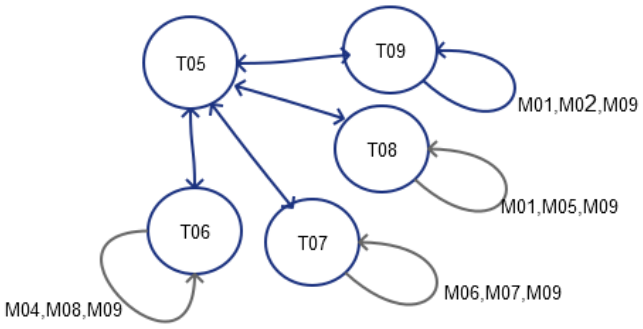
Gambar 3.46 Rancangan Antarmuka Pesan

3.2.5. Jaringan Semantik

Untuk mengetahui keterhubungan dari setiap tampilan yang dibuat dapat menggunakan jaringan semantik. Berikut adalah jaringan semantik pada Aplikasi interaktif pengolahan tubuh dapat dilihat pada gambar 3.47 sampai dengan 3.48.



Gambar 3.47 Jaringan semantic sisi client



Gambar 3.48 Jaringan semantic sisi server

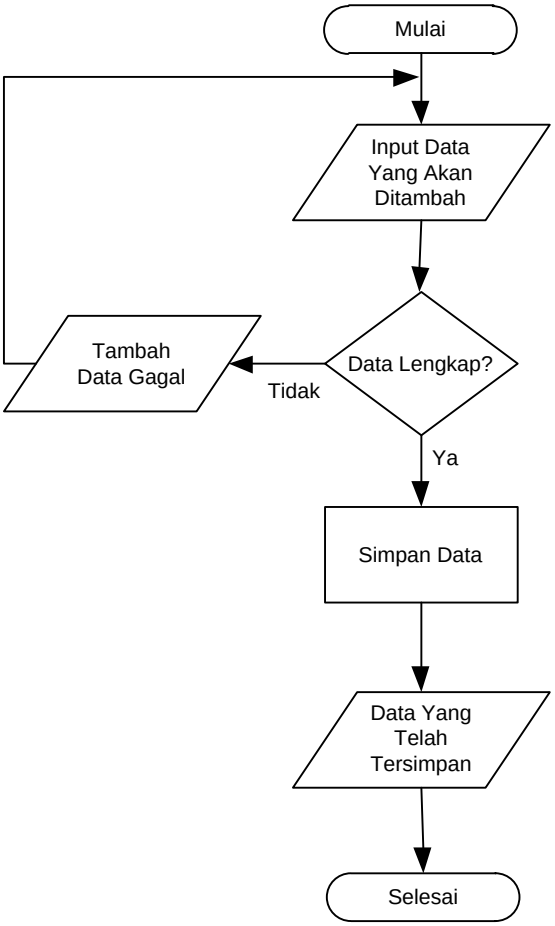
3.2.6. Perancangan Method

Perancangan komponen method merupakan perancangan yang dibuat setelah perancangan arsitektur menu, antarmuka dan jaringan semantik. Perancangan ini berfungsi untuk mendeskripsikan method-method yang berada di dalam aplikasi permainan. Method dapat dipanggil dengan menyertakan variabel, baik hanya satu variabel, banyak variabel atau bahkan tidak ada sama sekali. Adapun method-method yang terdapat dalam aplikasi hasil dari pengolahan tubuh yang akan dibangun dijelaskan melalui flowchart gambar 3.49- 3.51

a. Method Tambah data

Method tambah digunakan untuk melakukan inisialisasi terhadap setiap variabel, kelas dan objek-objek Method bisa dilihat pada gambar 3.49.

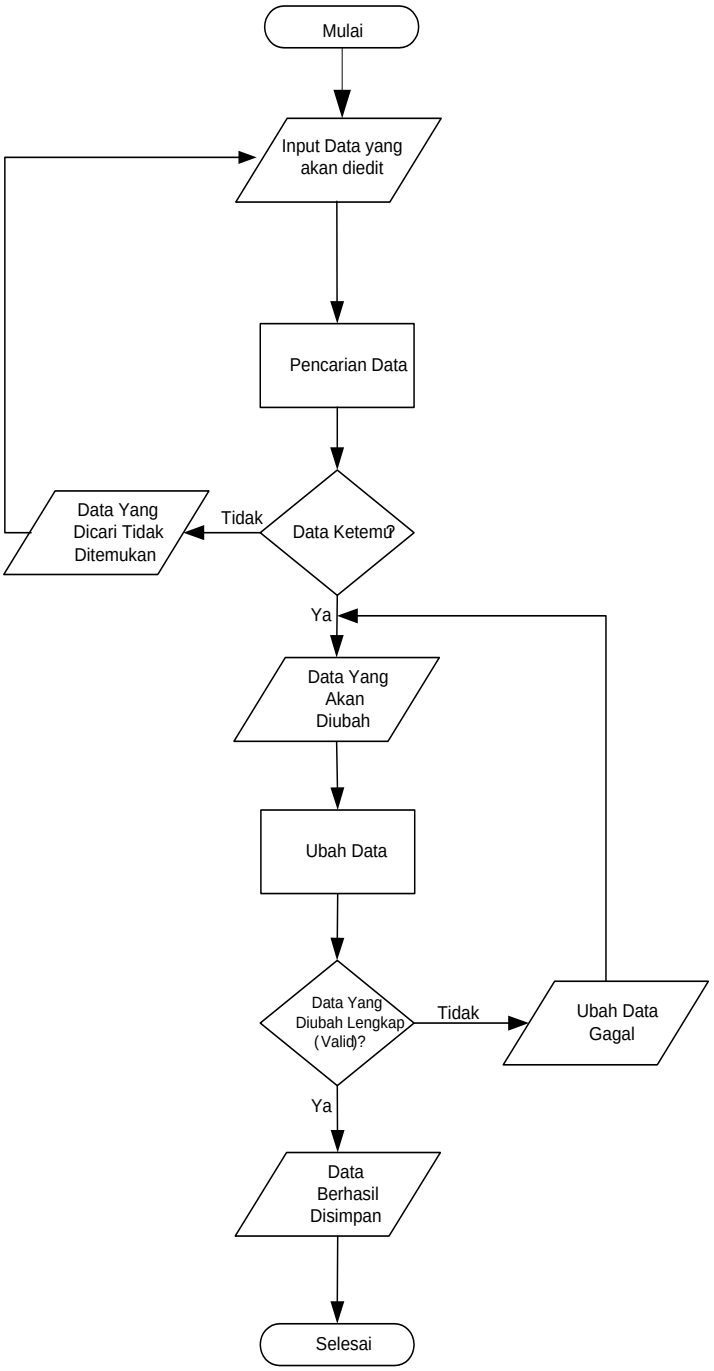
.



Gambar 3.47. Flowchart Method Input

b. Method Ubah data

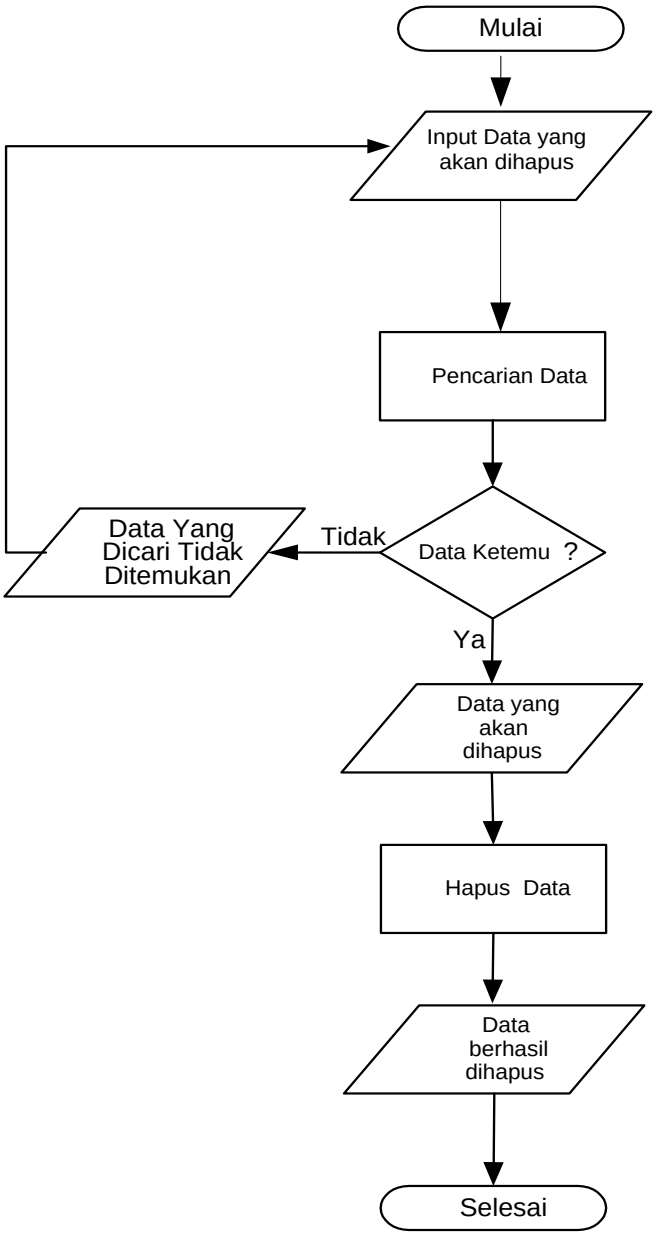
Method Update digunakan untuk melakukan pengupdatean terhadap kelas, variabel dan objek yang akan digunakan. Method bisa dilihat pada gambar 3.50



Gambar 3.50. Flowchart Method ubah data

c. Method Hapus data

Method Update digunakan untuk melakukan penghapusan terhadap kelas, variabel dan objek yang akan digunakan. Method bisa dilihat pada gambar 3.51



Gambar 3.51. Flowchart Method hapus data