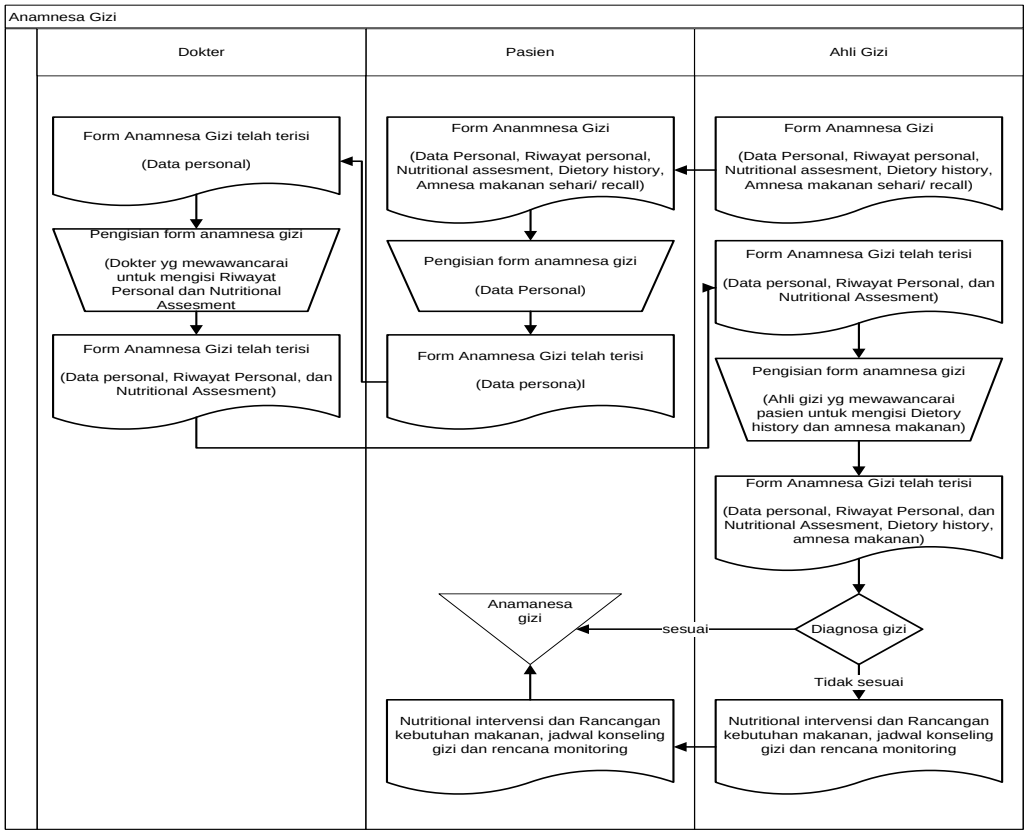
BAB III ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis dilakukan untuk mengidentifikasi tiap komponen yang terlibat dan mengelompokkannya menjadi sub-sub kecil. Hal ini bertujuan untuk mempermudah identifikasi dalam perancangan, pembangunan hingga antisipasi, sehingga mampu mengatasi kesalahan. Hal ini juga menjadi salah satu cara untuk membangan sebuah sistem yang kokoh dan sesuai kebutuhan.

3.1.1 Analisis Sistem berjalan

Sistem yang telah berjalan pada Amnesa gizi, ditunjukkan dengan flowmap dibawah ini:



Gambar III.1 flowmap anamnesa gizi

3.1.2 Analisis Masalah

Analisis masalah dilakukan untuk mengidentifikasi permasalahan-permasalahan yang kemudian menjadi sebuah persoalan yang dapat dijawab dan diselesaikan. Permasalahan yang akan dibahas dan diselesaiakan dalam sistem ini adalah

1. Perhitungan antrometri

   Perhitungan antrometri merupakan perhitungan untuk mendapatkan kesesuaian antara Tinggi Badan (TB), Berat Badan (BB) berdasarkan usia dan jenis kelamin.

2. Penentuan status dan kebutuhan gizi berdasarkan antrometri

   Status gizi akan dihitunga berdasarkan keadaan antrometri pasien yang akan disesuaikan kebutuhan gizinya hingga mendekati keadaan ideal antrometrinya.

3. Pengaturan pola makan normal

   Pengaturan pola makan normal yaitu untuk *user* normal (tidak mengidap diabetes) yang ingin mengolah menu hariannya sesuaikan dengan hasil dari poin 1 dan 2.

4. Pengaturan pola makan diabetes

   Pengaturan pola makan untuk penderita diabetes disesuaikan berdasarkan hasil dari poin 1, 2, dan aturan diet diabetes.

5. Informasi kandungan gizi menu kreasi

   Hal ini menghasilkan kandungan gizi dari sebuah menu kreasi/ menu yang dibuat sendiri oleh *user*.

Contoh penerapan secara manual:

Rina adalah seorang wanita berusia 22 tahun memiliki berat badan 54Kg dan tinggi badan 167cm, tidak memiliki kelainan pada penyakit (diabetes) dan aktifitas fisik sedang. Maka perhitungan yang dilakukan sebagai berikut:

1. BBN = 167 – 100 = 67 Kg
2. BBI = 67 – 6,7 = 60, 3 Kg
3. IMT $= \frac{54\ (\text{kg})}{1{,}67^2\ (\text{m})}$ = 19,36
4. BBE = 54 x 25 = 1350 kKal
5. Total Kalori = BBE + (BBE x 30%) = 1755 kKal
6. Karbo $= \frac{60\%\ \text{x}\ 1755}{4}$ = 263,25 gram
7. Lm $= \frac{25\%\ \text{x}\ 1755}{9}$ = 48,75 gram
8. Pr $= \frac{15\%\ \text{x}\ 1755}{4}$ = 65,81 gram

Berdasarkan perhitungan tersebut, makan dapat disimpulkan bahwa Rina:

1. Bila dibandingkan dengan berat BBN dan BBI maka BB Rina termasuk kurang
2. Nilai IMT sebesar 19,36, maka kecukupan gizi dan berat badannya termasuk normal. Bila Rina ingin mencapai berat badan ideal, maka dia haru menaikan berat badan hingga 60,3 Kg.
3. Kebutuhan total kalori sebesar 1755 kKal, dengan kebutuhan karbohidrat 263,25 gram, lemak 48,75 gram dan protein 65,81 gram.
4. Porsi makan yang sesuai dengan kebutuhannya dibagi berdasarkan jumlah kategori (per satuan penukar)

Tabel III.1 Penentuan porsi makan per kategori

Nasi / karbohidrat	Tempe/ protein nabati	Daging/ protein hewani	Sayuran	Buah	Gula	Minyak
4	3	2	3	4	2	5

3.1.3 Analisis Masukan

*Input*an yang dilakukana berasal dari alat *input keybord*. Bahasa yang digunakan adalaah bahasa Indonesia, tidak disingkat dan merupakan kata dasar.

Permasalahan utama dalam merancang sebuah aplikasi sistem cerdas dengan metode *natural language processing* adalah bagaimana membuat sebuah program yang mampu menerjemahkan *input*an *user* yang menggunakan bahasa alami (*natural language*) menjadi bahasa yang dapat dimengerti oleh komputer dan membuat program tersebut mampu memberikan respon kepada *user* menggunakan bahasa alami pula.

Masalah-masalah yang akan timbul dan harus ditangani oleh aplikasi ini adalah saat terjadi dialog dengan penggunanya yaitu sebagai berikut:

1. Kemampuan memproses bahasa alami

   Aplikasi ini akan berkomunikasi dengan komputer menggunakan bahasa yang digunakan oleh manusia dalam kehidupan sehari-hari atau bahasa alami. Maka, harus dibuat sebuah aplikasi yang mampu memberikan kemampuan kepada komputer untuk mengerti bahasa alami.

2. Ambiguitas kalimat

   Ambiguitas kalimat dapat terjadi saat satu kalimat atau lebih yang berbeda memiliki arti yang sama. Masalah utama yang menyebabkan terjadinya ambiguitas pada kalimat adalah saat ini penggunaan bahasa Indonesia tidak

selalu terpaku pada pola yang baku. Maksud dari pernyataan ini adalah dalam kehidupan pergaulan sehari-hari pada saat ini para pengguna bahasa Indonesia tidak terlalu terpaku pada susunan subjek, predikat, objek ataupun keterangan (S, P, O).

a. Ayam goreng
b. Ayam digoreng
c. Ayamnya digoreng

Dalam perbedaan *input*an tersebut memiliki satu maksud, yaitu *user* ingin mengetahui kandungan kalori dari ayam goreng.

3. Penanganan data yang cukup banyak pada bank data

Pintar atau tidaknya sebuah aplikasi ini turut ditentukan oleh banyaknya pengetahuan (*knowledge*) yang dimasukkan ke dalam bank data.

3.1.4 Analisis Proses

Proses yang berlangsung dalam sistem dibagi menjadi dua komponen utama, sebagai berikut:

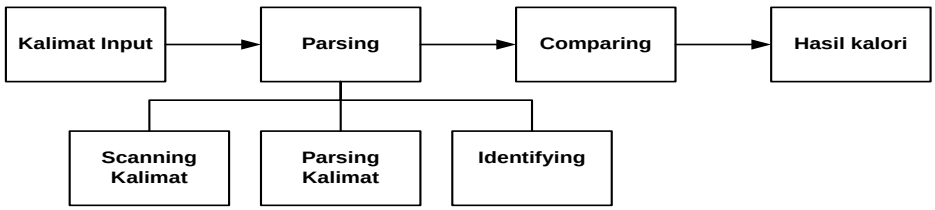
3.1.4.1 Analisis Pengolahan Bahasa Alami

Inti dari sistem NLP adalah parser. Parser adalah bagian dari program atau sistem yang membaca setiap kalimat, kata demi kata, untuk menentukan apa yang dimaksud. Ada 2 pendekatan yang bertolak belakang pada NLP. Pendekatan yang pertama mencoba untuk menggunakan semua informasi dalam satu kalimat seperti yang dilakukan manusia. Tujuan dari pendekatan ini adalah memungkinkan komputer melakukan percakapan. Pendekatan yang lain memungkinkan komputer menerima perintah dalam bahasa alami, tetapi cukup dengan cara mengekstrak

informasi yang esensial dari perintah tersebut. Salah satu aspek tersulit dalam mengembangkan sistem berbasis NLP adalah kompleksitas dan fleksibilitas dari bahasa manusia. Ketika membuat suatu *natural language processor*, maka perlu dibatasi tipe dari kalimat dan *grammar* yang diterima. Sehingga *processor* hanya akan memahami salah satu atau beberapa dari subset bahasa natural saja.

Dalam pembuatan sistem cerdas ini, *input* dari pengguna berupa *input* nonnumerik (kalimat bahasa alami). *Scanner* harus mengenali apakah *input* dari pengguna tersebut nonnumerik ataupun bukan kemudian memberikan *input* yang telah di-*scan* tersebut kepada *parser*. *Input* yang diterima oleh sistem berupa bahasa alami dipecah menjadi token-token penyusunnya yang berupa kata-kata penyusun kalimat *input* itu sendiri.

Pengolahan bahasa alami (NLP) memiliki presedur khusus dalam pengoperasiannya yang digambarkan dengan blok diagram berikut:



Gambar III.2 proses parsing

Blok diagram diatas menggambarkan garis besar pemprosesan bahasa yang dilakukan dalam aplikasi Nutrients Consult. Aplikasi berbasis web ini memiliki tujuan utama yaitu mengecek kandungan kalori dari setiap jenis makanan yang di-*input*-kan *user*. Penjelasan dari blok diagram diatas sebagai berikut:

1. Kalimat *input*

   Merupakan kalimat yang di-*input*-kan oleh *user*, berupa kumpulan kata yang merisikan bahan makanan ataupun menu makanan.

2. *Parsing*

   Parsing dilakukan untuk mengecek kesesuaian antara kalimat *input* dengan aturan produksi yang telah ditentukan. Proses *parsing* memilik 3 proses yaitu:

   a. *Scanning*

      Scanning (analisis leksikal) merupakan proses pemeriksaan pertama setelah dilakukan *input*-an. Pembacaan dilakukan satu persatu karakter dari kalimat yang di-*input*-kan. Hasil dari scanning berupa token-token yang kemudian dilanjutan pada proses *parsing*.

   b. *Parsing*

      Parsing (analisis sintaksis) berfungsi memeriksa kebenaran token-token yang telah dihasilkan dalam proses *scanner*. Pemeriksaan token disesuaikan dengan grammer yang telah ditentukan sebelumnya.

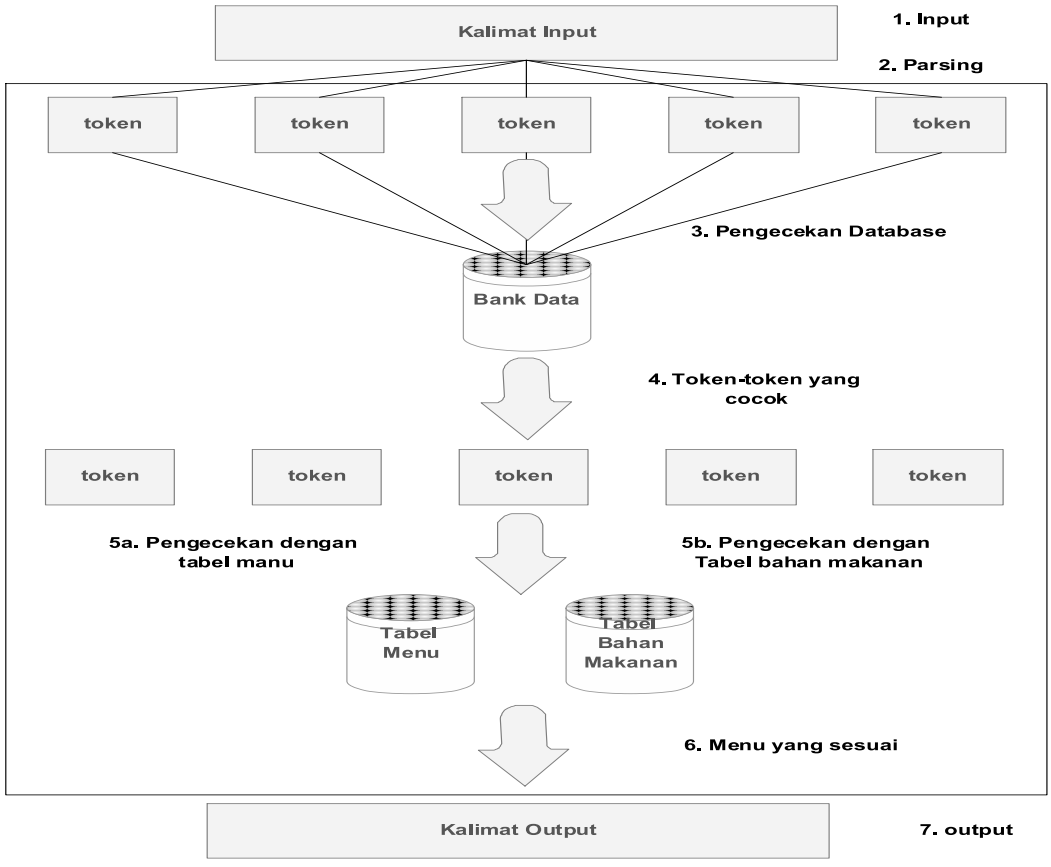
   c. *Identifying*

      Mengindentifikasi token-token yang telah sesuai dan diterima dengan *database* yang ada.

3. *Comparing*

Comparing dilakukan antara token dengan *database* sehingga menghasilkan keluaran berupa data yang dibutuhkan.

Setelah berupa token/ kata, maka sistem kemudian mencocokannya dengan isi bank data. Sistem berisi kumpulan algoritma dengan fungsi yang kompleks. Algoritma-algoritma yang terdapat di dalam sistem tersebut antara lain algoritma untuk *scanner, parser*, dan algoritma lainnya yang terkait dengan pengolahan *input* dari pengguna. Adapun gambaran proses *natural language processing* itu sendiri adalah sebagai berikut:



Gambar III.3 Global Parsing NLP

Penjelasan gambar:

1. *User* meng-*input*-kan menu yang akan di cek kandungannya
2. Menu di parsing perkata, menghasilkan token-token
3. Token-token dicocokan dengan *database* yang ada
4. Token-token yang cocok dengan *database*
5. Penyesuaian antara menu dan bahan makanan yang di-*input*-kan pada tabel menu dan bahan makanan
6. Penyesuaian menu dan perhitungan
7. *Output* berupa kandungan gizi dari menu yang di-*input*-kan

3.1.5 Analisis Proses Parser

Proses parser yang dilakukan pada *input*-an kalimat memiliki dua proses analisis utama, analisis leksikal dan analisis sintaksis. Kedua proses analisis ini memiliki peran yang cukup penting.

Analisis leksikal merupakan analisis pengecekan satu persatu karakter dari kalimat *input*an yang disesuaikan dengan aturan produksi yang ada. Hasil dari proses analisis ini disebut token. Token-token yang sesuai akan dilanjukan keproses analisis sintaksis. Proses analisis leksikal dikenal dengan proses *scanner*.

Analisis sintaksis merupakan analisis pengecekan token-token yang dihasilkan dari proses sebelumnya. Pengecekan token ini disesuaikan dengan grammer yang telah ditntukan sebelumnya. Apabila token telah sesuai maka akan dilanjutkan dengan proses identifikasi token dengan *database* yang ada. Analisis sintaksis dikenal dengan proses *parsing*.

3.1.5.1 Aturan produksi sintaks dan diagram sintaks

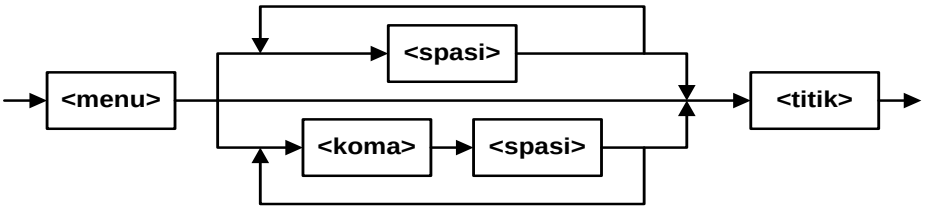
Atruran produksi untuk aplikasi *Anamnesa Gizi*, sebagai berikut:

Tabel III.2 Aturan Produksi

No.	α	::=	β
1.	<kalimat>	::=	(<menu>)$^{+}$ {<spasi>} | {<koma><spasi>} <titik>
2.	<menu>	::=	{<bahan_maknana>} | {<bahan_makanan><spasi><jenis_olahan>}
3.	<bahan_makanan>	::=	(<kata>)$^{+}$ | {<kata> <spasi> <kata>}
4.	<jenis_olahan>	::=	<kata>
5.	<kata>	::=	<token>
6.	<token>	::=	**(a...z)**$^{+}$
7.	<spasi>	::=	**'#'**
8.	<koma>	::=	**','**
9.	<titik>	::=	**'.'**

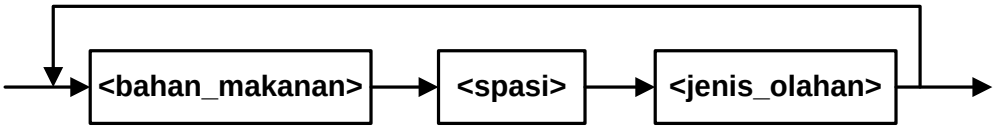
Untuk mempermudah memahami atuaran produksi BMF diatas, maka dijelaskan dengan diagram sintaks berikut ini.

<kalimat> ::= (<menu>)$^{+}$ {<spasi>} | {<koma><spasi>} <titik>



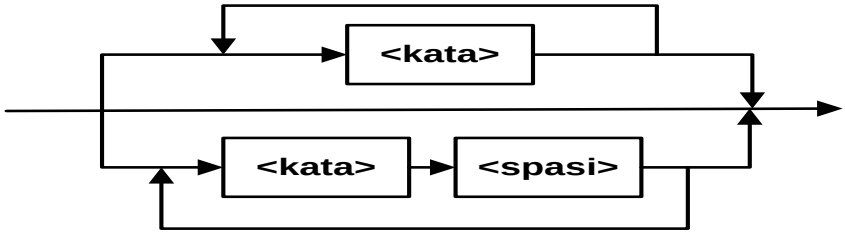
Gambar III.4 diagram sintaks kalimat

<menu> ::= <bahan_maknana> | <bahan_makanan><spasi><jenis_olahan>



Gambar III.5 diagram sintaks menu

<bahan_makanan> ::= (<kata>)$^{+}$ | {<kata> <spasi> <kata>}



Gambar III.6 diagram sintaks bahan_makanan

<jenis_olahan> ::= <kata>



Gambar III.7 diagram sintaks

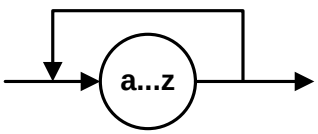
jenis_olahan

<kata> ::= <token>



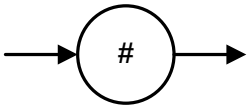
Gambar III.8 diagram sintaks kata

<token> ::= (a...z)$^{+}$



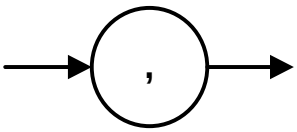
Gambar III.9 diagram sintaks token

<spasi> ::= '#'



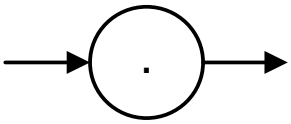
Gambar III.10 diagram sintaks spasi

<koma> ::= ','



Gambar III.11 diagram sintaks koma

<titik> ::= '.'

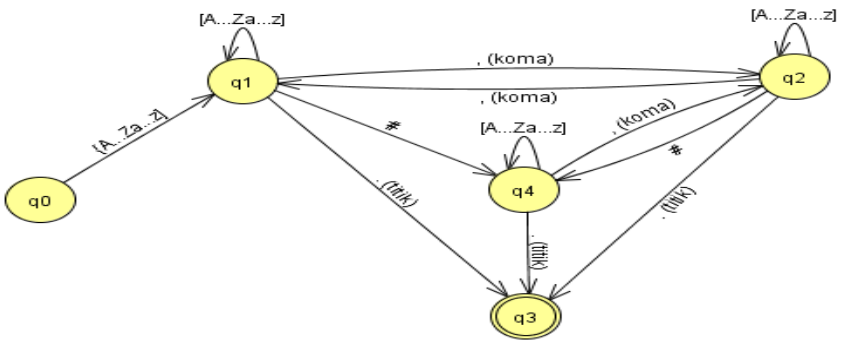


Gambar III.12 diagram sintaks titik

3.1.5.2 Diagram FSA (Finite State Automata) untuk analisis leksikal (scanner)

Aturan produksi untuk proses scanner berdasarkan FSA yang digambarkan dalam bentuk DFA dibawah ini:

Nasi, ayam goreng, tumis toge.



Gambar III.13 DFA untuk *scanner* NLP

Tuple yang terdapat dalam gambar diatas, sebagai berikut:

Q = {q0, q1, q2, q3, q4}

∑ = { a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, ‘,’, ‘.’, ‘#’}

S = {q0}

F = {q3}

Keterangan simbol yang digunakan sebagai *input*-an:

Tabel III.3 Keterangan simbol *input*

Simbol *input*	Keterangan
,	koma
.	titik
#	Pengganti spasi

Tabel transisi dari gambar DFA diatas, sebagai berikut:

Tabel III.4 tabel transisi DFA (part 1)

δ	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
q0	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1
q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1
q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2
q3															
q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4

Tabel III.5 tabel transisi DFA (part 2)

δ	p	q	r	s	t	u	v	w	x	y	z	.	,	#
q0	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1			
q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q1	q3	q2	q4
q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q2	q3	q1	q4
q3														
q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q4	q3	q2	q2

Untuk memastikan bahwa FSA yang dibuat telah sesuai, maka dilakukan pengecekan dengan *input* berikut ini:

Nasi, ayam goreng, tumis toge.

Diubah menjadi: Nasi,#ayam#goreng,#tumis#toge.

Hasil *tracing* dari kalimat diatas, dapat dilihat pada tabel berikut:

Tabel III.6 Transisi Hasil *Tracing*

Step	Transisi
1	δ(q0, N) = q1
2	δ(q1, a) = q1
3	δ(q1, s) = q1
4	δ(q1, i) = q1
5	δ(q1, ,) = q2

Step	Transisi
12	δ(q2, #) = q4
13	δ(q4, g) = q4
14	δ(q4, o) = q4
15	δ(q4, r) = q4
16	δ(q4, e) = q4

Step	Transisi
22	δ(q4, u) = q4
23	δ(q4, m) = q4
24	δ(q4, i) = q4
25	δ(q4, s) = q4
26	δ(q4, #) = q2

6	δ(q2, #) = q4
7	δ(q4, a) = q4
8	δ(q4, y) = q4
9	δ(q4, a) = q4
10	δ(q4, m) = q4
11	δ(q4, ,) = q2

17	δ(q4, n) = q4
18	δ(q4, g) = q4
19	δ(q4, ,) = q2
20	δ(q2, #) = q4
21	δ(q4, t) = q4

27	δ(q2, t) = q2
28	δ(q2, o) = q2
29	δ(q2, g) = q2
30	δ(q2, e) = q2
31	δ(q2, .) = **q3**

Berdasarkan *tabel* hasil *tracing* diatas menunjukkan bahwa kalimat *input*an diterima (*accepted*) karena *tracing* berawal dari q0 (*initial state*) dan berakhir di q3 (*final state*). Hasil *tracing* dapat dinyatakan dalam *regular expression* (RE) sebagai berikut.

$(([A..Z]+[a...z])^+,\#([A..Z]+[a...z])^+\#([A..Z]+[a...z])^+,\#([A..Z]+[a...z])^+([A..Z]+[a...z])^+.)$

3.1.5.3 Analisis parsing

Metode parsing yang digunakan untuk penelitian ini adalah metode parsing bottom-up. Penelusuran yang ddilakuakan mulai dari daun menuju puncak. Metode ini dapat menangani aturan produksi yang mengandung rekursif kiri dan hampir seluruh kelas bahasa CFG (*Context Free Grammer*). Metode ini akan mereduksi *string* masukan menjadi simbol awal dengan menggunakan aturan produksi yang ada.

Metode bottom-up parsing memiliki dua aksi yang dilakukan dalam proses parsing, yaitu *shift* (membaca *string* masukan berikutnya) dan *reduce* (mengubah *string* menjadi simbol variabel yang dapat menurunkan *string* tersebut). Hasil akhir dari metode ini berupa *accepted* (jika *string* masukan telah selesai dibacadan

state mampu mencapai state awal) atau *error* (jika *string* telah selesai dibaca namun tidak mencapai state awal, atau state awal telah tercapai namun masih terdapat *string* yang belum dibaca).

Contoh parsing untuk memeriksa string melakukan berikut ini:

Nasi,#ayam#goreng,#tumis#toge.

Berdasarkan aturan produksi pada tabel 3.1, simulasi parsing dengan metode bottom-up dapat dilihat dpada tabel 3.6 diberikut ini:

Tabel III.7 simulasi parsing

Stack	*Input* String	Action	Keterangan
	Nasi,#ayam#goreng,#tumis#toge.	Shift	
N	asi,#ayam#goreng,#tumis#toge.	Shift 3x	
Nasi	,#ayam#goreng,#tumis#toge.	Reduce	Token= Nasi Lihat aturan 5
<kata>	,#ayam#goreng,#tumis#toge.	Shift	Token= Nasi
<kata>,	#ayam#goreng,#tumis#toge.	Reduce	Lihat aturan 8
<menu> <koma>	#ayam#goreng,#tumis#toge.	shift	
<menu> <koma>#	ayam#goreng,#tumis#toge.	Reduce	Lihat aturan 7
<menu> <koma> <spasi>	ayam#goreng,#tumis#toge.	shift	
<menu> <koma> <spasi>a	yam#goreng,#tumis#toge.	Shift 3x	
<menu> <koma> <spasi>ayam	#goreng,#tumis#toge.	Reduce	Token = ayam Lihata aturan 5
<menu> <koma> <spasi><kata>	#goreng,#tumis#toge.	Shift	
<menu> <koma> <spasi><kata>#	goreng,#tumis#toge.	Reduce	Lihat aturan 7

Stack	*Input* String	Action	Keterangan
<menu> <koma><spasi> <kata><spasi>	goreng,#tumis#toge.	shift	
<menu> <koma> <spasi><kata><spasi> g	oreng,#tumis#toge.	Shift 5x	
<menu> <koma><spasi> <kata><spasi> goreng	,#tumis#toge.	Reduce	Token = goreng Lihat aturan 5
<menu> <koma><spasi> <menu>	,#tumis#toge.	Shift	
<menu> <koma><spasi ><menu>,	#tumis#toge.	Reduce	
<menu> <koma><spasi> <menu><koma>	#tumis#toge.	Shift	
<menu> <koma><spasi> <menu>#	tumis#toge.	Reduce	
<menu> <koma><spasi> <menu> <koma>	tumis#toge.	Shift 5x	
<menu> <koma><spasi> <menu> <koma>tumis	#toge.	Reduce	Token = tumis Lihat aturan 5
<menu> <koma><spasi> <menu> <koma> <kata>	#toge.	Shift	
<menu> <koma><spasi> <menu> <koma> <kata>#	toge.	Reduce	Lihat aturan 7
<menu> <koma><spasi> <menu> <koma> <kata><spasi>	toge.	Shift 4x	
<menu> <koma><spasi> <menu> <koma> <kata><spasi>toge	.	Reduce	Token = tumis Lihat aturan 5
<menu> <koma><spasi> <menu> <koma> <bahan_makan> <jenis_olahan>.		Shift Reduce	Lihat aturan 9
<menu> <koma> <spasi><menu> <koma> <bahan_makan> <jenis_olahan> <titik>		Reduce	Lihat aturan 1
<kalimat>			

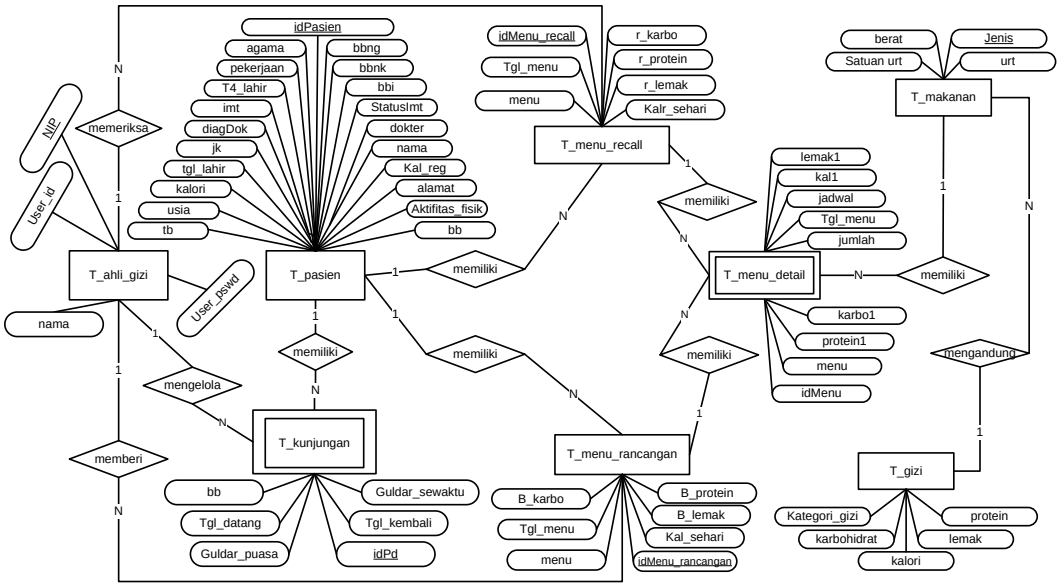
Stack	*Input* String	Action	Keterangan
Finish			

3.1.6 Analisis Keluaran

Hasil keluaran sistem ini berupa menu makan, bila *user* ingin menunya dipilihka. Apabila *user* tidak sesuai dengan menu yang ditentukan sistem dan ingin memilih menunya sendir maka *output* berupa kandungan gizi dari manu yang dipilih *user*. Apabila terdapat kelebihan kalori dari yang dibutuhkan maka *user* diberi beberapa alternatif untuk mengimbanginya. Hal tersebut dapat berupa olah raga ringan atau penyesuaiaa menu pada waktu makan berikutnya.

3.1.7 Analisis Basis Data

Analisis basis data merupakan sebuah proses analisis tentang aliran data yang diolah dan digunakan dalam *database*. Dalam proses ini direpresentasikan atribut tiap data dan aliran data. Basis data merupakan kumpulan data yang saling berkaitan satu dengan lainnya yang direalisasikan dengan *relation key* yang digambarkan dalam *entity relationship diagram (ERD)*. Adapun ERD yang terbentuk sebagai berikut:

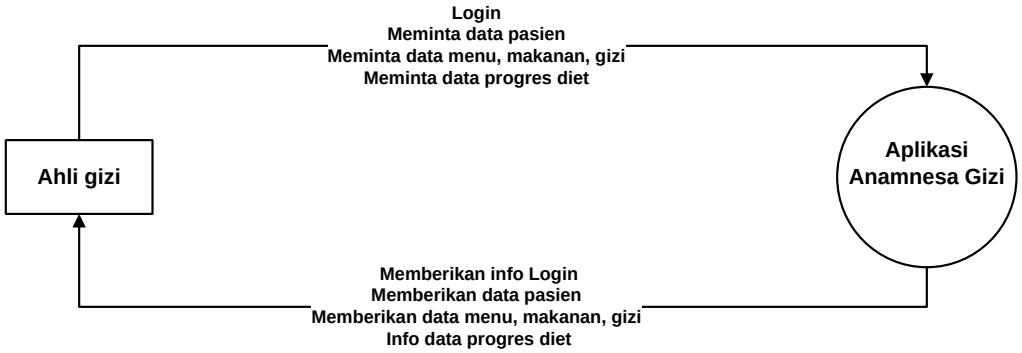


Gambar III.14 *Entity Relationship Diagram*

3.1.8 Analisis Kebutuhan Fungsional

3.1.8.1 Diagram konteks

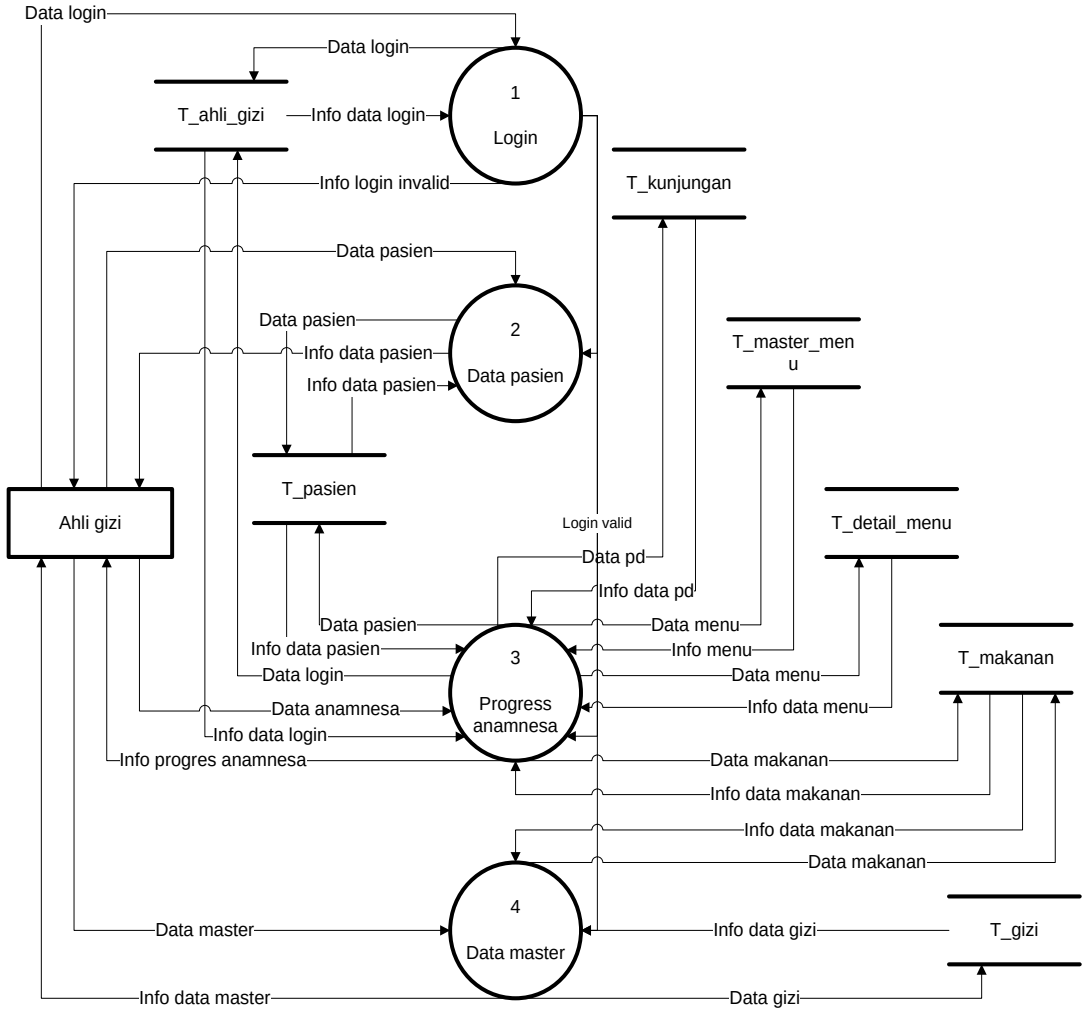
Diagram Konteks adalah diagram yang menggambarkan secara umum yang menjadi masukan, proses dan keluaran yang terjadi pada sebuah sistem. Diagram konteks untuk perangkat lunak yang akan dibangun sebagai berikut :



Gambar III.15 Diagram konteks anamnesa Gizi

3.1.8.2 DFD level 1

DFD Level 1, merupakan turunan pertama dari diagram konteks sebelumnya. Pada DFD level 1, menggambarkan runtutan proses utama yang terdapat pada aplikasi. Umumnya DFD Level 1 ini menjadi over view dari menu dasar yang tersedia pada aplikasi.

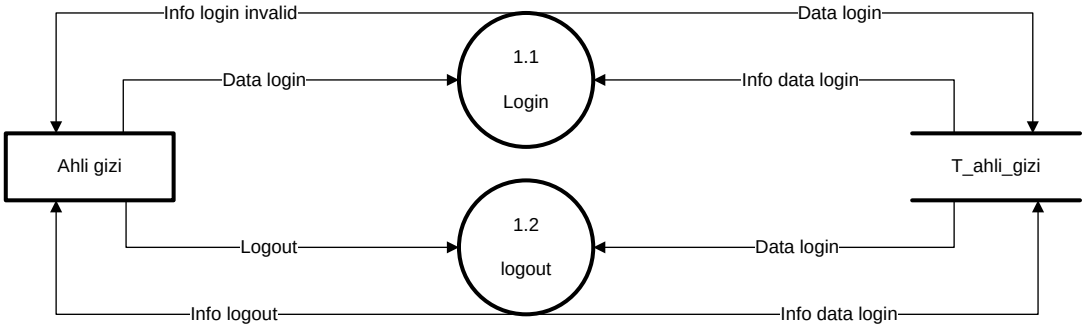


Gambar III.16 DFD Level 1, Anamnesa Gizi

3.1.8.3 DFD Level 2

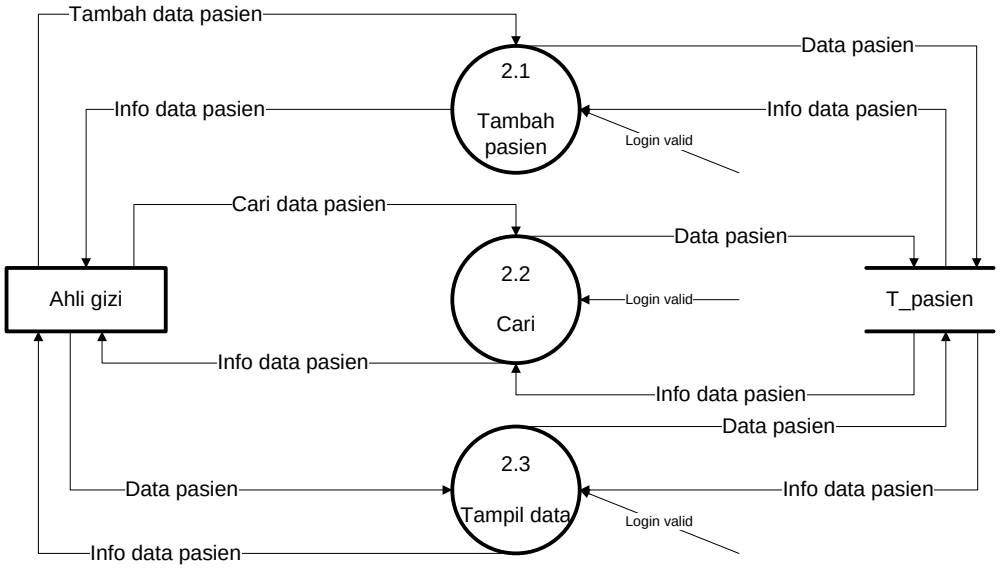
DFD Level 2, merupakan turukan ke-2 dari diagram konteks atau turunan ke-1 dari DFD Level 1. Pada DFD Level 2, dirincikan beberapa proses yang masih memiliki turunan. Sehingga hasil analisis dapat lebih termodelkan.

1. Login



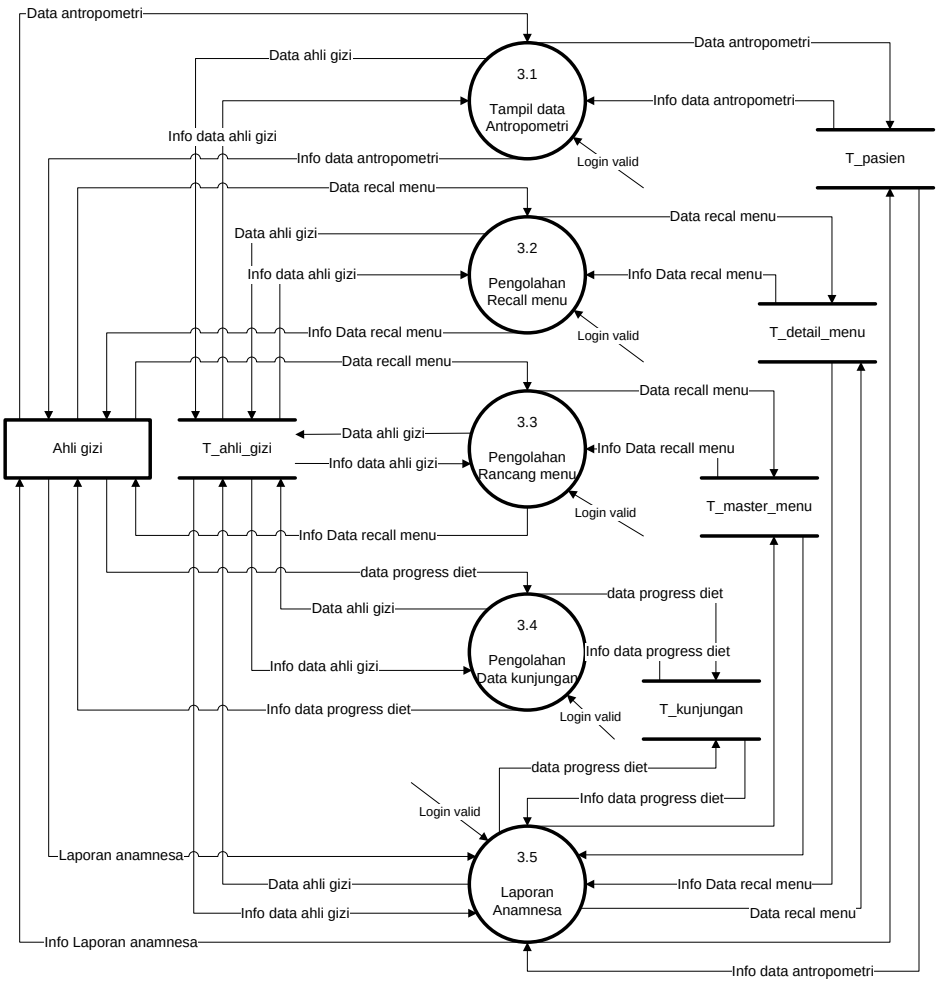
Gambar III.17 DFD Level 2, Proses Login

2. Data pasien



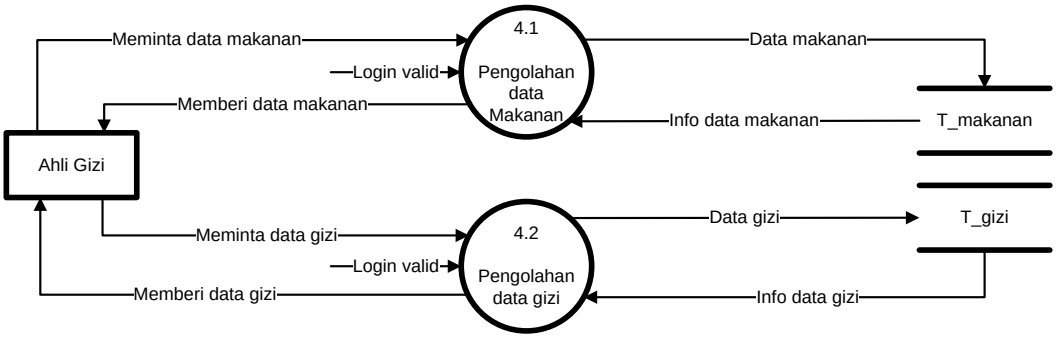
Gambar III.18 DFD Level 2, Proses Pengolahan Data Pasien

3. Progress anamnesa



Gambar III.19 DFD Level 2, Pengolahanprogress anamnesa

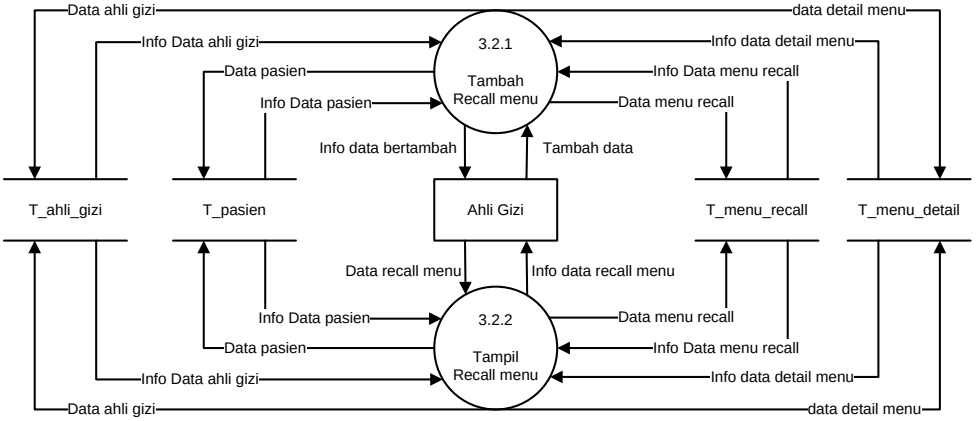
4. Data master



Gambar III.20 DFD Level 2, Pengolahan Data Master

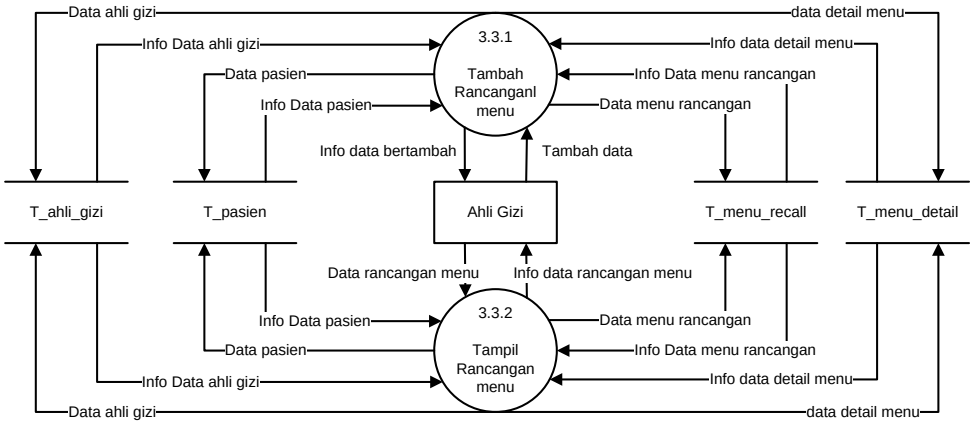
3.1.8.4 DFD Level 3

1. *Recall* menu



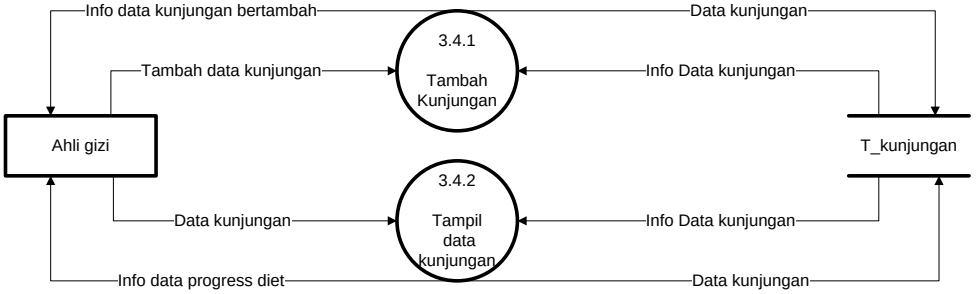
Gambar III.21 DFD Level 3, proses *recall* menu

2. Rancang menu



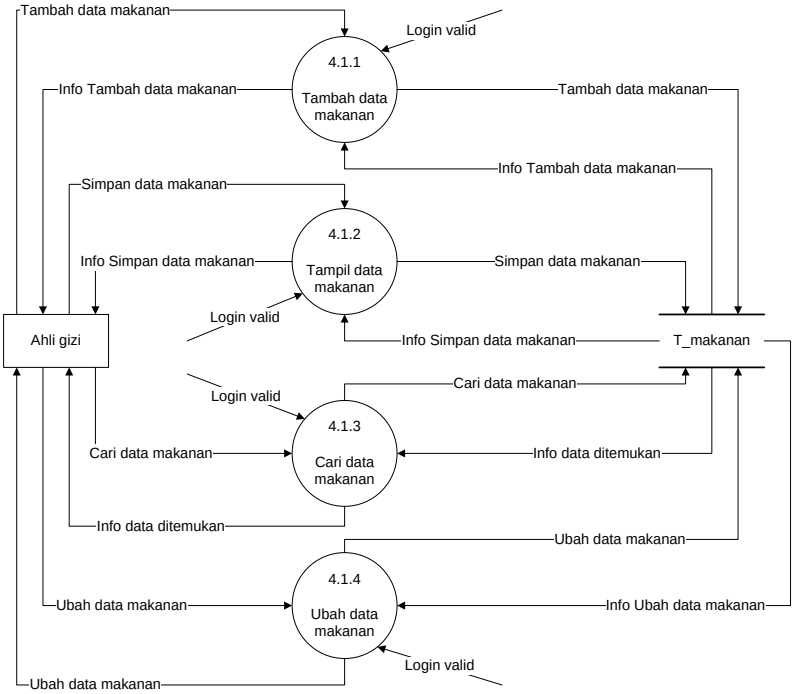
Gambar III.22 DFD Level 3, proses rancang menu

3. Data kunjungan



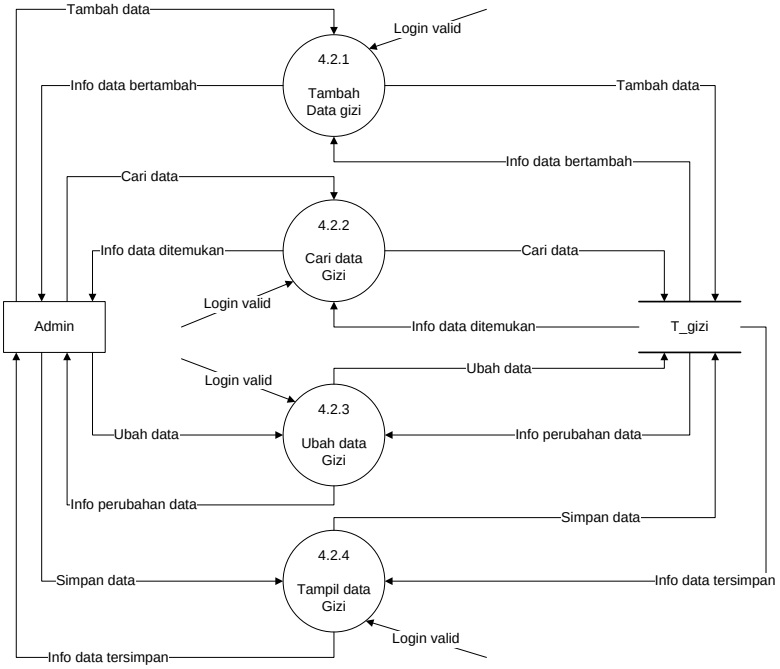
Gambar III.23 DFD Level 3, proses kunjungan

4. Pengolahan data makanan



Gambar III.24 DFD Level 3, proses pengolahan data makanan

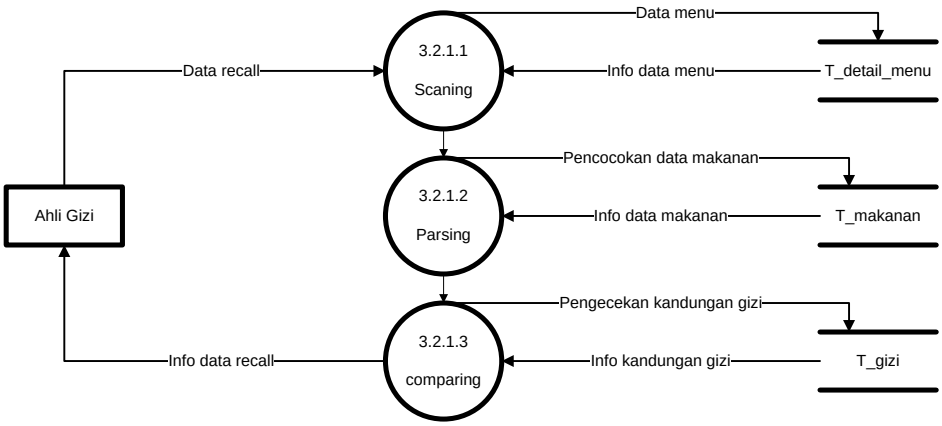
5. Pengolahan data gizi



Gambar III.25 DFD Level 3, proses pengolahan data gizi

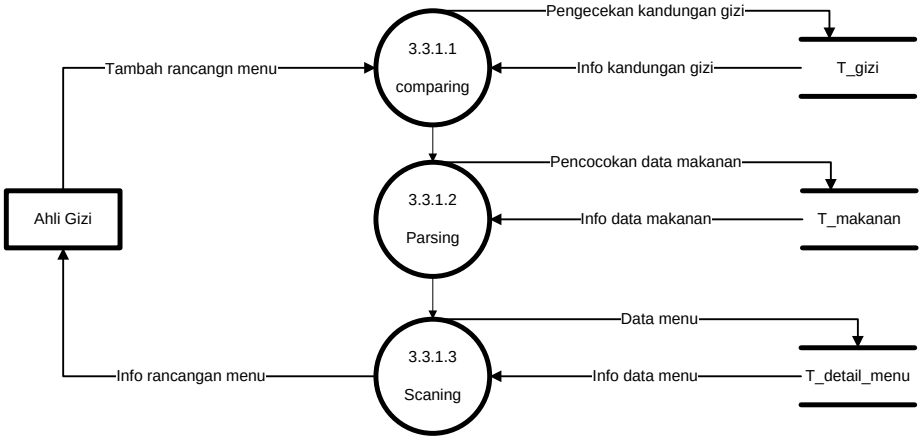
3.1.8.5 DFD Level 4

1. Tambah *recall* menu



Gambar III.26 DFD Level 4, proses tambah *recall* menu

2. Tambah rancangan menu



Gambar III.27 DFD Level 3, proses tambah rancangan menu

3.1.8.6 Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan logika dari proses-proses yang digambarkan pada DFD. Spesifikasi proses pada aplikasi ini, sebagai berikut:

Tabel III.8 Spesifikasi proses

NO.	PROSES	KETERANGAN
1.	No. Proses	1.1
	Nama	Login
	Source	Ahli gizi
	Input	ID *user* dan *Pasword*
	Output	Form home
	Destination	Ahli gizi
	Logika proses	1. Bila ID user kosong, muncul pesan konfirmasi bahwa ID user kosong dan harus diisi 2. Bila ID user tidak sesuai, maka muncul pesan konfirmassi bahwa ID user belum terdaftar 3. Bila Password kosong, muncul pesan konfirmasi bahwa pasword kosong dan harus diisi 4. Bila password tidak sesuia dengan ID user, maka muncul pesan konfirmasi bahwa pasword salah 5. Bila ID user dan pasword benar maka form home tampil
2.	No. Proses	1.2
	Nama	*Logout*
	Source	Ahli gizi
	Input	-
	Output	*Close* program
	Destination	Ahli gizi
	Logika proses	Bila tombol *logout* ditekan maka semua proses ditutup dan program keluar
3.	No. Proses	2.1
	Nama	Tambah pasien
	Source	Ahli gizi
	Input	Data pasien
	Output	Info data pasien berhasil disimpan
	Destination	Ahli gizi
	Logika proses	1. Saat tombol tambah di tekan maka *field* data pasien aktif (keadaan awal *readonly*) 2. Saat tombol simpan di tekan, maka dilakukan pengecekan pada field-field yang wajib diisi telah terisi atau belum, bila belu maka keluar pesan

NO.	PROSES	KETERANGAN
		konfirmasi bahawa field tersebut masih kosong dan harus diisi (set fokus pada field yang kosong) 3. Bila semua data yang dimasukkan telah terisi dan sesuai maka data akan tersimpan dimemori dan IdPasien diberikan secara otomatis 4. Namun bila proses dibatalkan, keluar pesan konfirmasi apakan pembatalan benar dilakukan?, bila iy maka seluruh field yang terisi dikosongkan dan field kembali pada keadaan semula (readonly)
4.	No. Proses	2.2
	Nama	Cari pasien
	Source	Ahli gizi
	Input	Id pasien atau nama pasien
	Output	Data pasien yang dicari
	Destination	Ahli gizi
	Logika proses	1. Memilih jenis pencarian (berdasarkan nama atau id pasien) 2. Mengisikan data berdasarkan jenis pencarian 3. Bila data ditemukan maka data akan ditampilkan pada field data pasien yang disediakan dan pointer di daftar tabel pasien menunjuk data yang ditemukan 4. Bila tidak ditemukan maka data keluar pesan konfirmasi bahwa data tidak ditemukan
5.	No. Proses	2.3
	Nama	Tampil data pasien
	Source	Ahli gizi
	Input	-
	Output	Data pasien dalam bentuk tabel
	Destination	Ahli gizi
	Logika proses	1. Data pasien ditampilkan dalam bentuk tabel secara keseluruhan 2. Data pada baris yang aktif ditampilkan dalam field-field data pasien yang disediakan

NO.	PROSES	KETERANGAN
6.	No. Proses	3.1
	Nama	Tampil data antropometri
	Source	Ahli gizi
	Input	Id pasien
	Output	Data antropometri pasien
	Destination	Ahli gizi
	Logika proses	Saat form proses anamnesa tampil, maka tab pertama yang tebuka adalah data antropometri dari pasien
7.	No. Proses	3.5
	Nama	Laporan
	Source	Ahli gizi
	Input	Id pasien
	Output	Priview data pasien dan hasil rancangan menu
	Destination	Ahli gizi
	Logika proses	Saat tombol laporan dipilih maka menampilkan laporan yang siap untuk di cetak
8.	No. Proses	3.2.2
	Nama	Tampil recall menu
	Source	Ahli gizi
	Input	Id pasien
	Output	Data recall dalam bentuk tabel dan detail
	Destination	Ahli gizi
	Logika proses	Saat tombol pencarian dilakukan maka tab recall menempilkan seluruh data history recall pasien yang dipilih dalam bentuk tabel dan data yang terakhir di tampilkan dalam bentuk detail dalam sebuah field
9.	No. Proses	3.3.2
	Nama	Tampil rancangan menu
	Source	Ahli gizi
	Input	Id Pasien
	Output	Data rancangan menu dalam bentuk tabel dan detail
	Destination	Ahli gizi
	Logika proses	Saat tombol pencarian dilakukan maka pada tab rancangan menu menempilkan seluruh data history rancangan menu pasien yang dipilih dalam bentuk

NO.	PROSES	KETERANGAN
		tabel dan data yang terakhir di tampilkan dalam bentuk detail dalam sebuah field
10.	No. Proses	3.4.1
	Nama	Tambah kunjungan
	Source	Ahli gizi
	Input	Data kunjungan
	Output	Info data kunjungan telah bertambah dan berhasil disimpan
	Destination	Ahli gizi
	Logika proses	1. Tombol tambah di klik maka seluruh field aktif, data tanggal kedatangan dan datang kembali telah terisi secara otomatis 2. Data berat badan pada kunjungan pertama diambil dari data pasien 3. Saat proses penyimpanan terdapat validasi untuk menyatakan field benar-benar kosong atau lupa terisi (pada field gula darah sewaktu dan gula darah puasa) bila benar-benar kosong/ tidak ada data dilakukan pengisian defauld bernilai nol (0) 4. Billa dataa lupa dimasukkan maka setfokus pada field yang kosong 5. Saat semua data terisis dengan benar maka data disimpan dan keluar pesan konfirmasi bahwa data berhasil disimpan
11.	No. Proses	3.4.2
	Nama	Tampil data kunjungan
	Source	Ahli gizi
	Input	ID pasien
	Output	Data kunjungan dalam bentuk tabel dan detail field
	Destination	Ahli gizi
	Logika proses	Data kunjungan ditampilkan dalam bentuk tabel dan field pada tab data kunjungan
12.	No. Proses	4.1.1
	Nama	Tambah data makanan
	Source	Ahli gizi

NO.	PROSES	KETERANGAN
	Input	Data makanan
	Output	Info data makanan bertambah
	Destination	Ahli gizi
	Logika proses	1. Saat tombol tambah ditekan maka field aktif dan disediakan tempat pada memori, sehingga pengisian data dapat dilakukan 2. Saat tombol simpan ditekan maka ada pengecekan pada field-field yang kosong, set fokus pada field yang kosong 3. Bila semua data terisi dan benar maka data akan dismpan pada memory yang telah disediakan dan keluar info bahwa data telah tersimpan 4. Bila data dibatalkan akan keluar pesan konfirmasi apakah proses akan dibatalkan, bila iya maka seluruh field akan dikosongkan dan penyimpanan dibatalkan
13.	No. Proses	4.1.2
	Nama	Tampil data makanan
	Source	Ahli gizi
	Input	-
	Output	Data makanan ditampilkan
	Destination	Ahli gizi
	Logika proses	Data makanan ditampilkan dalam bentuk tabel
14.	No. Proses	4.1.3
	Nama	Cari data makanan
	Source	Ahli gizi
	Input	Jenis makanan
	Output	Info Data makanan yang dicari atau pesan informasi bahwa data tidak ditemukan
	Destination	Ahli gizi
	Logika proses	1. Tombol cari ditekan, maka keluar box pencarian berdasarkan jenis makanan 2. Bila jenis makanan ditemukan maka data makanan yang dicari akan ditampilkan 3. Bila tidak maka keluar pesan konfirmasi bahwa data tidak ditemukan

NO.	PROSES	KETERANGAN
15.	No. Proses	4.1.4
	Nama	Ubah data makanan
	Source	Ahli gizi
	Input	Perubahan data makanan
	Output	Info data makanan berhasil diubah
	Destination	Ahli gizi
	Logika proses	1. Saat tombol ubah ditekan maka field aktif dan dapat diubah 2. Saat tombol simpan ditekan maka ada pengecekan pada field-field yang kosong, set fokus pada field yang kosong 3. Bila semua data terisi dan benar maka data akan dismpan pada memory yang telah disediakan dan keluar info bahwa data telah tersimpan 4. Bila data dibatalkan akan keluar pesan konfirmasi apakah proses akan dibatalkan, bila iya maka seluruh field akan dikosongkan dan penyimpanan dibatalkan
16.	No. Proses	4.2.1
	Nama	Tambah data gizi
	Source	Ahli gizi
	Input	Data gizi
	Output	Info data gizi berhasil disimpan
	Destination	Ahli gizi
	Logika proses	1. Saat tombol tambah ditekan maka field aktif dan disediakan tempat pada memori, sehingga pengisian data dapat dilakukan 2. Saat tombol simpan ditekan maka ada pengecekan pada field-field yang kosong, set fokus pada field yang kosong 3. Bila semua data terisi dan benar maka data akan dismpan pada memory yang telah disediakan dan keluar info bahwa data telah tersimpan 4. Bila data dibatalkan akan keluar pesan konfirmasi apakah proses akan dibatalkan, bila iya maka seluruh field akan dikosongkan dan penyimpanan

NO.	PROSES	KETERANGAN
		dibatalkan
17.	No. Proses	4.2.2
	Nama	Cari data gizi
	Source	Ahli gizi
	Input	Kategori gizi
	Output	Data gizi yang dicari atan pesan informasi bahwa data yang dicari tidak ditemukan
	Destination	Ahli gizi
	Logika proses	1. Tombol cari ditekan, maka keluar box pencarian berdasarkan jenis makanan 2. Bila jenis makanan ditemukan maka data makanan yang dicari akan ditampilkan 3. Bila tidak maka keluar pesan konfirmasi bahwa data tidak ditemukan
18.	No. Proses	4.2.3
	Nama	Ubah data gizi
	Source	Ahli gizi
	Input	Data gizi yang akan diubah
	Output	Info data gizi berhasil berubah
	Destination	Ahli gizi
	Logika proses	1. Saat tombol ubah ditekan maka field aktif dan dapat diubah 2. Saat tombol simpan ditekan maka ada pengecekan pada field-field yang kosong, set fokus pada field yang kosong 3. Bila semua data terisi dan benar maka data akan dismpan pada memory yang telah disediakan dan keluar info bahwa data telah tersimpan 4. Bila data dibatalkan akan keluar pesan konfirmasi apakah proses akan dibatalkan, bila iya maka seluruh field akan dikosongkan dan penyimpanan dibatalkan
19.	No. Proses	4.2.4
	Nama	Tampil data gizi
	Source	Ahli gizi

NO.	PROSES	KETERANGAN
	Input	-
	Output	Data gizi dalam bantuk tabel
	Destination	Ahli gizi
	Logika proses	Data gizi ditampilkan dalam bentuk tabel
20.	No. Proses	3.2.1.1
	Nama	*scanning*
	Source	Ahli gizi
	Input	Jenis dan jumlah makanan
	Output	Token
	Destination	parsing
	Logika proses	1. Jenis dan jumlah makanan yang dimasukkan di cek seseuaiannya dengan data yang terdapat dibatabase, menjadi token 2. Jumlah makanan yang diinputkan dicek dengan jumlah makanan yang terdapat didatabase
21.	No. Proses	3.2.1.2
	Nama	*Parsing*
	Source	Ahli gizi
	Input	Token
	Output	Parsing token
	Destination	Comparing
	Logika proses	Token yang sesuai di periksa gramernya dilanjutkan pada prose comparing
22.	No. Proses	3.2.1.3
	Nama	*comparing*
	Source	Ahli gizi
	Input	Token hasil parsing
	Output	Nilai kandungan gizi
	Destination	Ahli gizi
	Logika proses	1. Pemerisaan token terhadap jumlah yang dimasukkan dengan jumlah satuan berat pada database 2. Dilakukan perbandingan antara berat inputan dan berat satuan yang kemudian akan

NO.	PROSES	KETERANGAN
		dilkalikan dengan nilai kandungan gizi per satuan berat pada database 3. Nilai total perhitungan dihasilkan
23.	No. Proses	3.3.1.1
	Nama	*Comparing*
	Source	Ahli gizi
	Input	Nilai kalori harian
	Output	Nilai protein, lemak, dan karbohidrat
	Destination	Parsing
	Logika proses	1. Dilakukan pencacahan nilai dari kalori total yang dibutuhkan menjadi nilai kalori per kandungan gizi (lemak , protein dan karbohidrat) 2. Pencarian jumlah dan kategori makanan yang dibutuhkan
24.	No. Proses	3.3.1.2
	Nama	*Parsing*
	Source	Ahli gizi
	Input	Jumlah makanan yang dibutuhkan
	Output	Jenis makanan
	Destination	*Scanning*
	Logika proses	1. Jumlah tiap kategori disesuaikan dengan jumlah satuan jenis makanan 2. Jenis makanan yang dibutuhkan ditampilkan dalam bentuk token dan penyesuaian gramernya
25.	No. Proses	3.3.1.1
	Nama	*Scanning*
	Source	Ahli gizi
	Input	Token hasil parsing
	Output	Data menu sehari
	Destination	Ahli gizi
	Logika proses	Token yang telah seuai gramernya ditampilkan dalam bantuk menu sehari

3.1.9 Analisis Kebutuhan Nonfungsional

3.1.9.1 Analisis Perangkat Keras

Perangkat keras adalah seluruh komponen atau unsur peralatan yang digunakan untuk menunjang pembangunan aplikasi. Adapun perangkat keras yang digunakan untuk pemakaian aplikasi ini secara optimal memerlukan spesifikasi minimum komputer sebagai berikut :

1. Processor 1.7 GHz
2. Memori DDR 1 Gb
3. Hardisk 40 Gb
4. Monitor 15 Inc

3.1.9.2 Analisis Perangkat Lunak

3.1.9.3 Analisis Pengguna

Pengguna pada sistem ini dibedakan berdasarkan hak akses yang diberikan, yaitu admin dan member. Pendaftaran member dapat dilakukan secara langsung melalui website yang diakses sedangkan admin telah didaftarkan saat sistem dibangun. Karakteristik rinci *user* dapat dilihat dari *tabel* dibawah ini:

Tabel III.9 analisi pengguna

Pengguna	Tanggung Jawab	Hak Akses	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman	Jenis Pelatihan
Ahli gizi	Menjalankan dan meng-update data pada aplikasi	Keseluruhan proses pada aplikasi	Min D3 ilmu gizi atau jurusan terkait	Mampu mengoprasikan komputer dan memahami ilmu gizi	Pengetahuan seputar ilmu gizi, ahli gizi, konsultan gizi	Pelatihan penggunaan aplikasi

3.2 Perancangan

3.2.1 Perancangan Basis Data

Perancangan basis data menggambarkan rancangan dari kumpulan data-data yang saling terhubung, terlibat dan tersimpan pada aplikasi secara bersama.

3.2.2 Kamus Data

Kamus data merupakan perincian komponen data dari setiap data yang telibat dan digambarkan dalam DFD. Berikut ini adalah kamus data yang digunakan dalam aplikasi.

Tabel III.10 Kamus data

No.	Nama Aliran Data	Data ahli gizi
1.	Where Used/ How Used	Proses 1.1, proses 1.2, proses 3.1, proses 3.2, proses 3.3, proses 3.4, proses 3.5
	Deskripsi	Berisi data ahli gizi
	Struktur Data	NIP, nama, user_id, user_pswd
	NIP Nama User_id User_pswd	{a...z | A...Z| 0...9} {a...z | A...Z} {a...z | A...Z| 0...9} {a...z | A...Z| 0...9}
	Nama Aliran Data	Data pasien
2.	Where Used/ How Used	Proses 2.1, proses 2.2, proses 2.3, proses 2.4, proses 3.1
	Deskripsi	Berrisi data pasien
	Struktur Data	idPasien, nama, alamat, jk, usia, bb, tb, imt, kalori, kal_reg, aktifitas fisik, tgl_lahir, bbi, bbnk, bbng, statusImt, t4_lahir, agama, giag_dok, dokter, pekerjaan
	idPasien nama alamat jk usia bb	{0...9} {a...z | A...Z} {a...z | A...Z| 0...9} {‘laki-laki’, ‘perempuan’} {0...9} {0...9}

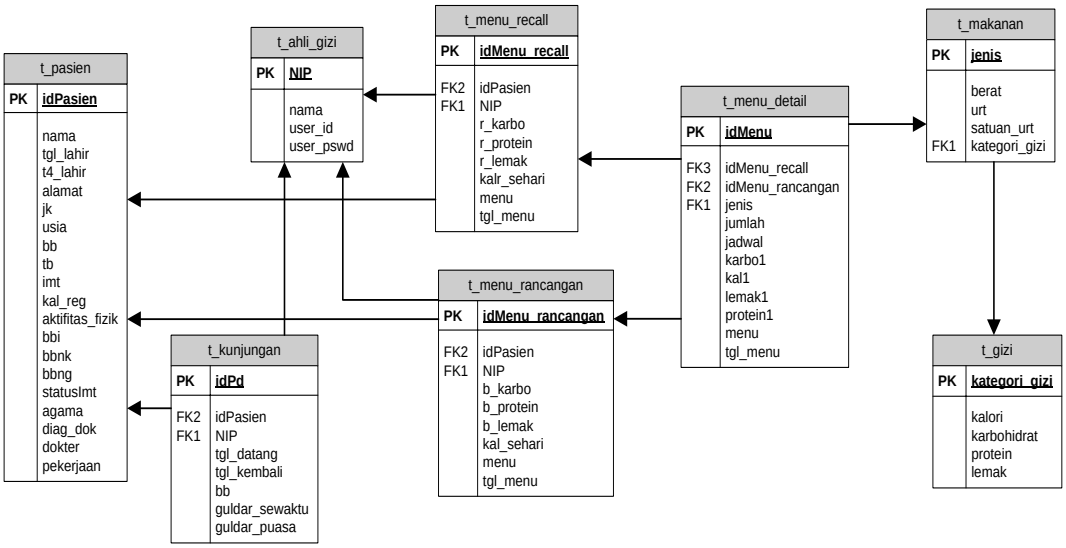
	tb imt kalori kal_reg aktifitas fisik tgl_lahir bbi bbnk bbng statusImt t4_lahir agama diag_dok dokter pekerjaan	{0...9} {0...9} {0...9} {0...9} {‘bed rest’, ‘ringan’, ‘sedang’, ‘berat’} Date {0...9} {0...9} {0...9} {‘berat badan kurang’, ‘berat badan normal’, ‘berat badan lebih’, ‘beresiko obesitas’, ‘obesitas1’, ‘obesitas2’} {a...z | A...Z} {‘islam’, ‘Protestan’, ‘katolik’, ‘hindu’, ‘budha’, ‘konghucu’} {a...z | A...Z} {a...z | A...Z} {a...z | A...Z}
	Nama Aliran Data	Data kunjungan
3.	Where Used/ How Used	Proses 3.4.1, proses 3.4.2
	Deskripsi	Berisi data kunjungan
	Struktur Data	idPd, tgl_datang, tgl_kembali, bb, guldar_sewaktu, guldar_puasa
	idPd tgl_datang tgl_kembali bb guldar_sewaktu guldar_puasa	{0...9} Date Date {0...9} {0...9} {0...9}
	Nama Aliran Data	Data menu recall
4.	Where Used/ How Used	Proses 3.2.1 dan proses 3.2.2
	Deskripsi	Berisi data recall
	Struktur Data	idMenu_recall, r_karbo, r_protein, r_lemak, kalr_sehari, menu, tgl_menu
	idMenu_recall r_karbo r_protein r_lemak kalr_sehari	{0...9} {0...9} {0...9} {0...9} {0...9}

	menu tgl_menu	{a...z | A...Z| 0...9} date
	Nama Aliran Data	Data menu rancangan
5.	Where Used/ How Used	Proses 3.3.1 dan proses 3.3.2
	Deskripsi	Berisi data rancangan menu
	Struktur Data	idMenu_rancangan, b_karbo, b_protein, b_lemak, kal_sehari, menu, tgl_menu
	idMenu_rancangan b_karbo b_protein b_lemak kal_sehari menu tgl_menu	{0...9} {0...9} {0...9} {0...9} {0...9} {a...z | A...Z| 0...9} date
	Nama Aliran Data	Data menu detail
6.	Where Used/ How Used	Proses 3.2.1.1 dan proses 3.3.1.3
	Deskripsi	Berisi data detail menu
	Struktur Data	idMenu, jadwal, karbo1, protein1, lemak1, kal1, menu, tgl_menu, jumlah
	IdMenu Jadwal Karbo1 Protein1 Lemak1 Kal1 Menu Tgl_menu Jumlah	{0...9} {‘sarapan’, ‘snack pagi’, ‘makan siang’, ‘snack siang’, ‘makan malam’, ‘snack malam’} {0...9} {0...9} {0...9} {0...9} {a...z | A...Z| 0...9} Date {0...9}
	Nama Aliran Data	Data makanan
7.	Where Used/ How Used	Proses 3.2.1.2, proses 3.3.1.2, proses 4.1.1, proses 4.1.2, proses 4.1.3, proses 4.1.4
	Deskripsi	Berisi data makanan
	Struktur Data	Jenis, berat, urt, satuan_urt
	Jenis Berat Urt Satuan urt	{a...z | A...Z} {0...9} {0...9} {‘buah’, ‘buah kecil’, ‘buah sedang’ ,’buah besar’,

		‘batang’, ‘batang kecil’, ‘batang sedang’, ‘batang besar’, ‘biji’, ‘biji kecil’, ‘biji sedang’, ‘biji besar’, ‘butir’, ‘ekor’, ‘ekor kecil’, ‘ekor sedang’, ‘ekor besar’, ‘gelas’, ‘potong', 'potong kecil', 'potong sedang', 'potong besar', 'sendok makan', 'sendok teh', 'sekehendak', '-'}
	Nama Aliran Data	Data gizi
8.	Where Used/ How Used	Proses 3.2.3, proses 3.3.1.1, proses 4.2.1, proses 4.2.2, proses 4.2.3, proses 4.2.4
	Deskripsi	Berisi data gizi
	Struktur Data	Ketegori_gizi, kalori, karbohidrat, protein, lemak
	Kategori_gizi Kalori karbohidrat Protein lemak	{a...z | A...Z} {0..9} {0..9} {0..9} {0..9}
	Nama Aliran Data	Data antropometri
9.	Where Used/ How Used	Proses 3.1
	Deskripsi	Data anamnesa merupakan data hasil proses dari data pasien dan terdapat dalam tabel pasien
	Struktur Data	Tb, bbi, bbnk, bbng, kalori, imt, statusImt
	Tb Bbi Bbnk Bbng Kaloti Imt StatusImt	{0...9} {0...9} {0...9} {0...9} {0...9} {0...9} {‘berat badan kurang’, ‘berat badan normal’, ‘berat badan lebih’, ‘beresiko obesitas’, ‘obesitas1’, ‘obesitas2’}

3.2.3 Skema Relasi

Skema relasi menggambarkan relasi yang terjadi antar tabel dan dilengkapi dengan ketengan dan atribut tiap tabel.



Gambar III.28 Skema relasi

3.2.4 Perancangan Struktur Tabel

Perancangan skruktur tabel berfungsi menggambarkan komponen tiap tabel secara lebih rinci. Kompenen-komponen yang dijelaskan berupa nama kolom, tipe data, ukuran serta keterangannya. Keterangan yang dibuat seputar key (Primary key, foreign key) maupun keterangan lain yang menjelaskan tiap kolom yang ada dalam tabel (tabel *reference* bila terdapat foreign key). Perancangan struktur tabel yang terdapat dalam Anamnesa gizi, sebagai berikut.

Tabel III.11 t_ahli_gizi

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	NIP	Char	18	Primary	
2.	Nama	String	50		
3.	*User*_id	Char	20		
4.	*User*_pswd	Char	8		

Tabel III.12 t_pasien

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	idPasien	int	8	Primary	
2.	nama	string	50		
3.	alamat	string	250		
4.	jk	enum			‘Laki-laki’, ‘Perempuan’
5.	usia	int	11		
6.	bb		3		
7.	tb	int	3		
8.	imt	float	5		
9.	kalori	float	5		
10.	Kal_reg	int	5		
11.	aktifitas_fisik	enum			‘bed rest’, ‘ringan’, ‘sedang’, ‘berat’, ‘sangat berat’
12.	tgl_lahir	date			
13.	Bbi	float			
14.	Bbnk	float			
15.	Bbng	float			
16.	StatusImt	enum			'berat badan kurang', 'berat badan normal', berat badan lebih', 'beresiko obesitas', 'obesitas 1', 'obesitas 2'
17.	T4_lahir	String	20		
18.	agama	enum			‘Islam’, ‘Katolik’, ‘Budha’, ‘Protestan’, ‘Hindu’, ‘Konghucu’
19.	Diag_dok	String	37		
20.	Dokter	String	40		
21.	Pekertjaan	string	27		

Tabel III.13 t_kunjungan

NO.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	idPd	int	7	Primary	
2.	idPasien	int	8	Foreign	*Reference* t_pasien
3.	NIP	Char	18	Foreign	*Reference* t_ahli_gizi

NO.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
4.	tgl_datang	date			
5.	tgl_kembali	date			
6.	Bb	int	4		
7.	Guldar_sewaktu	Int	3		
8.	Guldar_puasa	Int	3		

Tabel III.14 t_menu_recall

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	idMenu_recall	int	4	Primary	
2.	idPasien	int	8	Foreign	*reference* t_pasien
3.	NIP	Char	18	foreign	*Reference* t_Ahli_gizi
4.	r_karbo	Float			
5.	r_protein	float			
6.	r_lemak	Float			
7.	Kalr_sehari	float			
8.	Menu				
9.	Tgl_menu	date			

Tabel III.15 t_menu_rancangan

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	idMenu_rancangan	int	4	Primary	
2.	idPasien	int	8	Foreign	*reference* t_pasien
3.	NIP	Char	18	foreign	*Reference* t_Ahli_gizi
4.	B_karbo	Float			
5.	B_protein	Float			
6.	B_lemak	Float			
7.	Kal_sehari	float			
8.	Menu				
9.	Tgl_menu	date			

Tabel III.16 t_menu_detail

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	idMenu	int	4	Primary	
2.	idMenu_recall	int	4	Foreign	*reference*

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
					t_Menu_recall
3.	jenis	char	35	Foreign	Reference t_makanan
4.	jadwal	enum			‘sarapan’, ‘*snack* pagi’, ‘makan siang’, ‘*snack* siang’, ‘makan malam’, ‘*snack* malam’
5.	Karbo1	float			
6.	Protein1	Float			
7.	Lemak1	Float			
8.	Kal1	float			
9.	Menu				
10.	Tgl_menu	date			
11.	jumlah	float			
12.	idMenu_rancangan	int	4	Foreign	Reference t_menu_rancangan

Tabel III.17 t_makanan

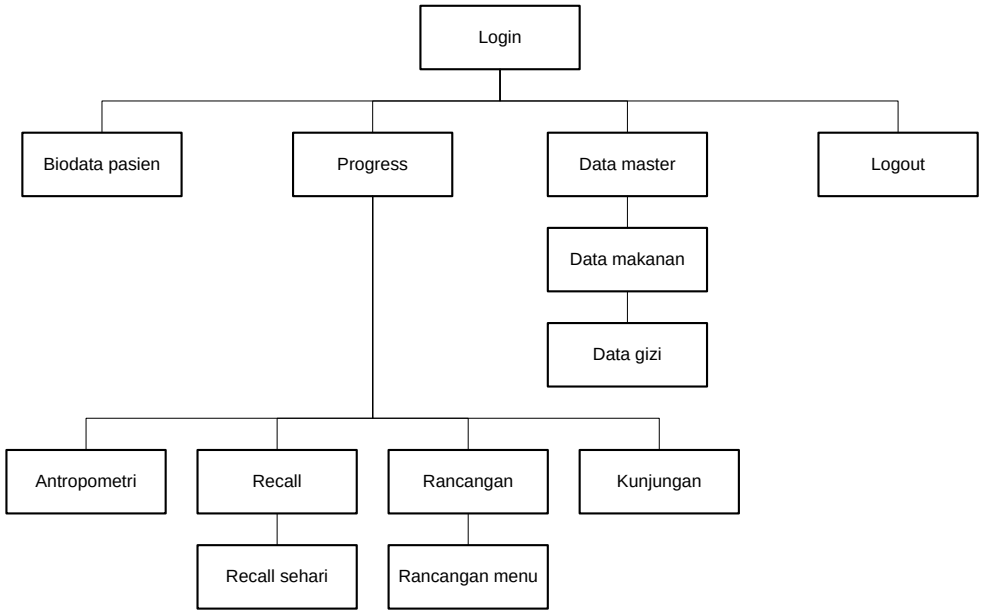
No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	jenis	String	35	PK	
2.	Kategori_gizi	String	35	FK	*Reference*
3.	berat	Int	3		
4.	urt	Float			
5.	satuan_urt	enum			‘buah’, ‘buah kecil’, ‘buah sedang’ ,’buah besar’, ‘batang’, ‘batang kecil’, ‘batang sedang’, ‘batang besar’, ‘biji’, ‘biji kecil’, ‘biji sedang’, ‘biji besar’, ‘butir’, ‘ekor’, ‘ekor kecil’, ‘ekor sedang’, ‘ekor besar’, ‘gelas’, ‘potong', 'potong kecil', 'potong sedang', 'potong besar', 'sendok makan', 'sendok teh', 'sekehendak', '-'

Tabel III.18 t_gizi

No.	Nama Kolom	Tipe	Ukuran	Key	Keterangan
1.	Kategori_gizi	String	35	PK	
2.	Kalori	Int	4		
3.	Karbohidrat	Int	4		
4.	Protein	Int	4		
5.	Lemak	int	4		

3.2.5 Perancangan Arsitektur

3.2.5.1 Perancangan Struktur Menu

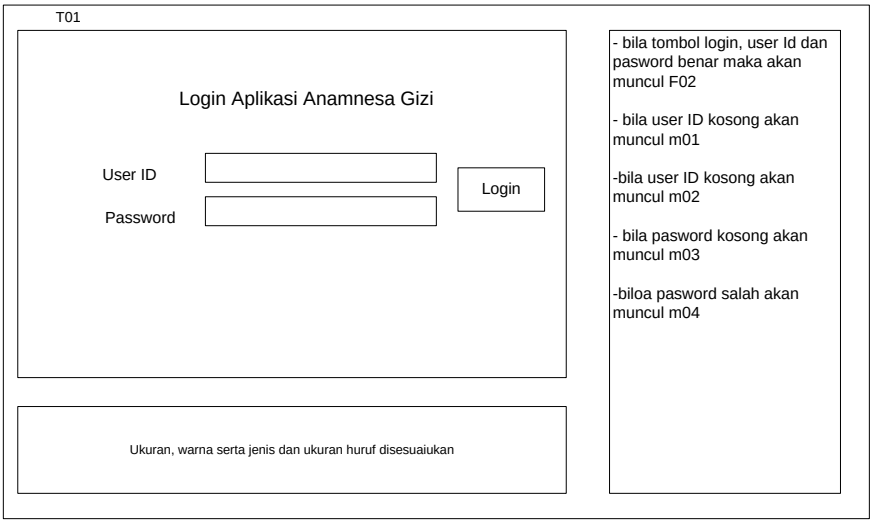


Gambar III.29 Perancangan struktur menu

3.2.5.2 Perancangan Antar Muka

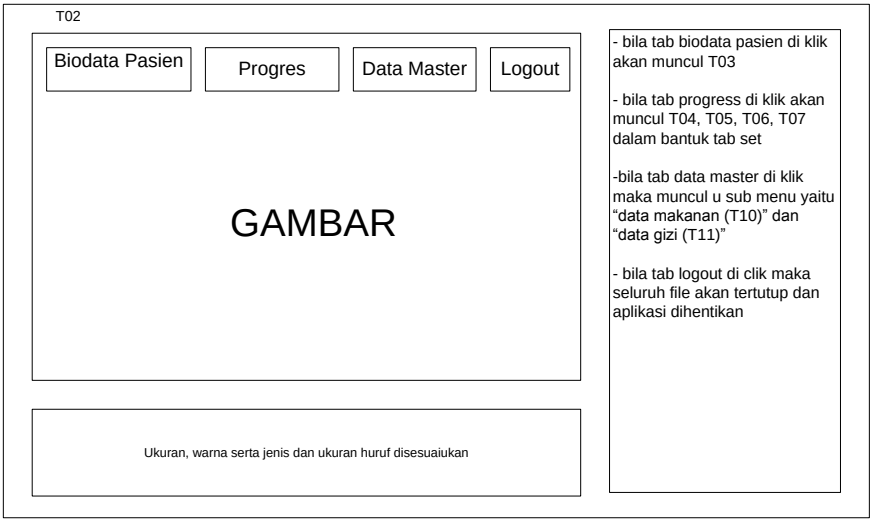
3.2.5.2.1 Perancangan Antar Muka Penunjang

1. Perancangan antar muka Login (T01)



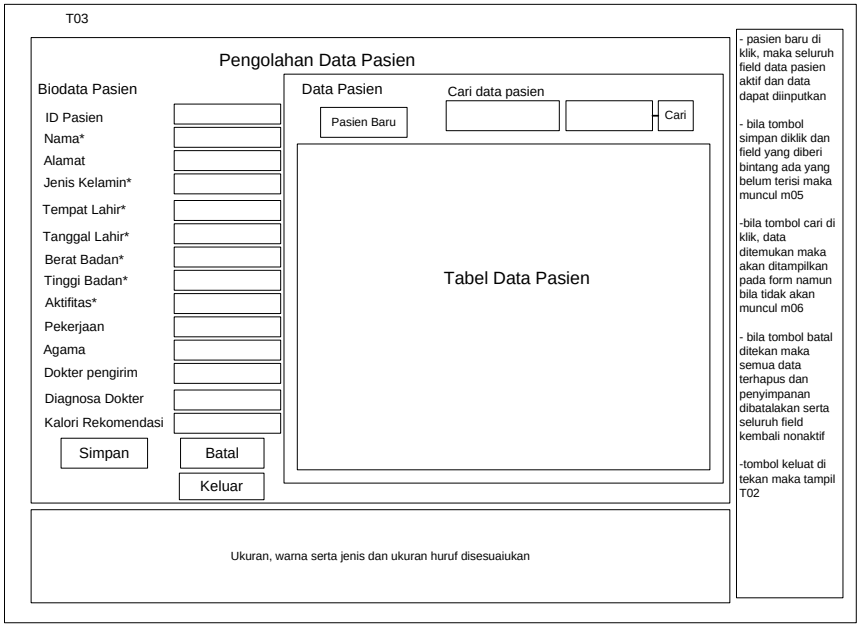
Gambar III.30 Perancangan antar muka Login

2. Perancangan antar muka home (T02)



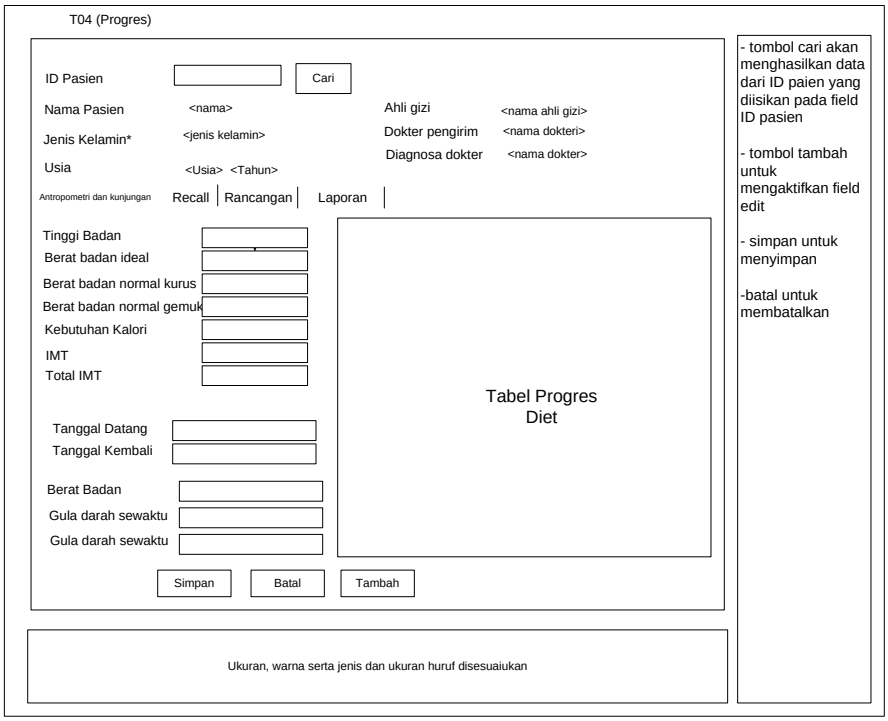
Gambar III.31 Perancangan antar muka home

3. Perancangan antar muka pengolahan data pasien (T03)



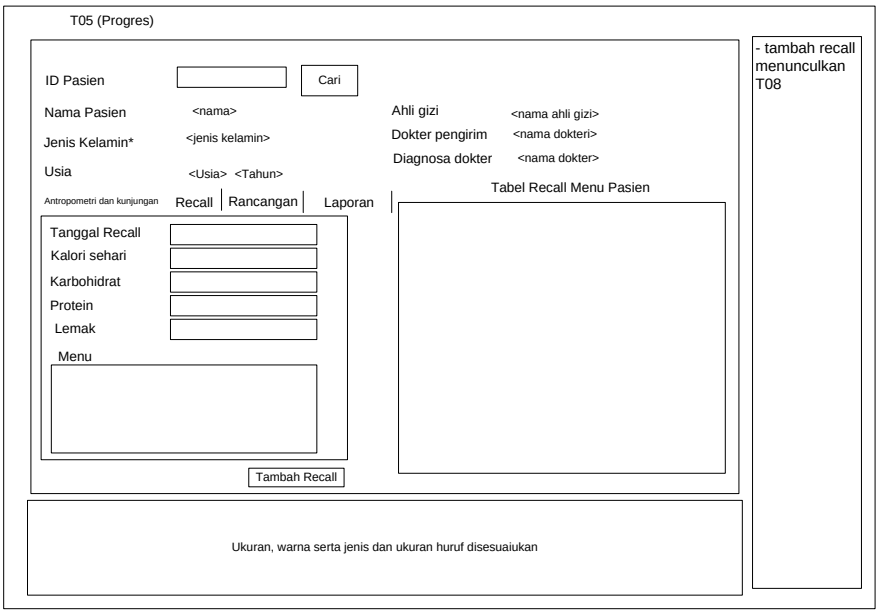
Gambar III.32 Perancangan antar muka pengolahan data pasien

4. Perancangan antar muka proses anamnesa-antropometri (T04)



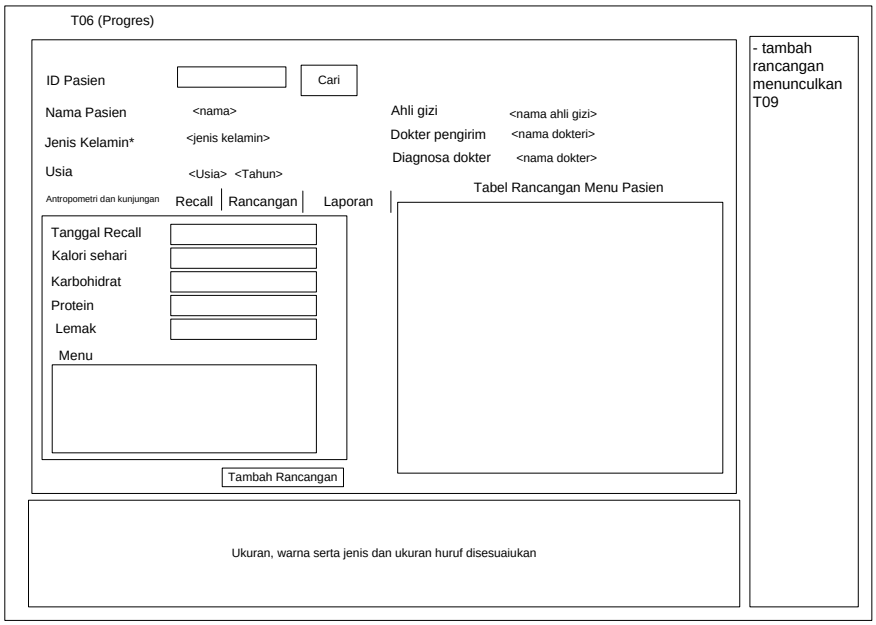
Gambar III.33 Perancangan antar muka proses anamnesa-antropometri

5. Perancangan antar muka proses anamnesa-Recall (T05)



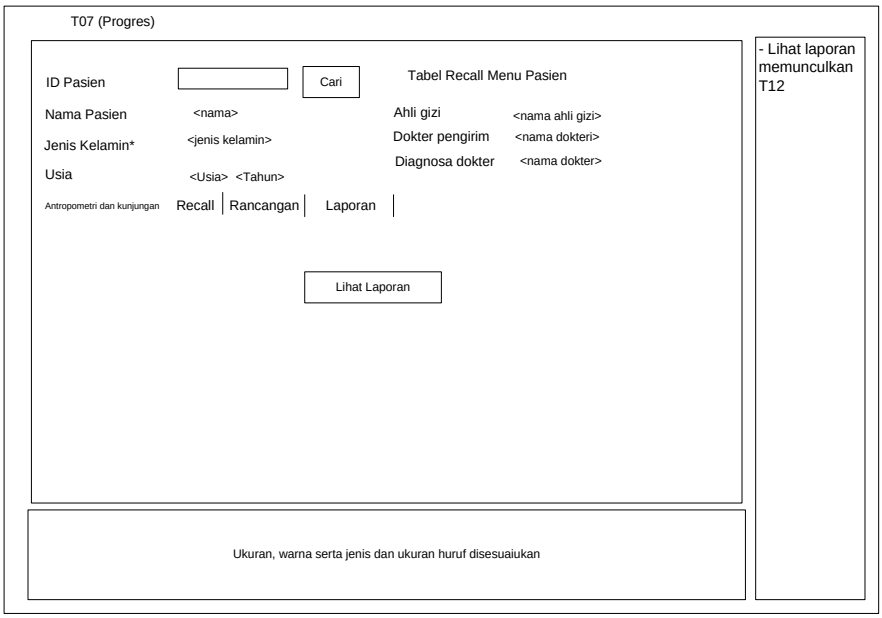
Gambar III.34 Perancangan antar muka proses anamnesa-Recall

6. Perancangan antar muka proses anamnesa-Rancangan (T06)



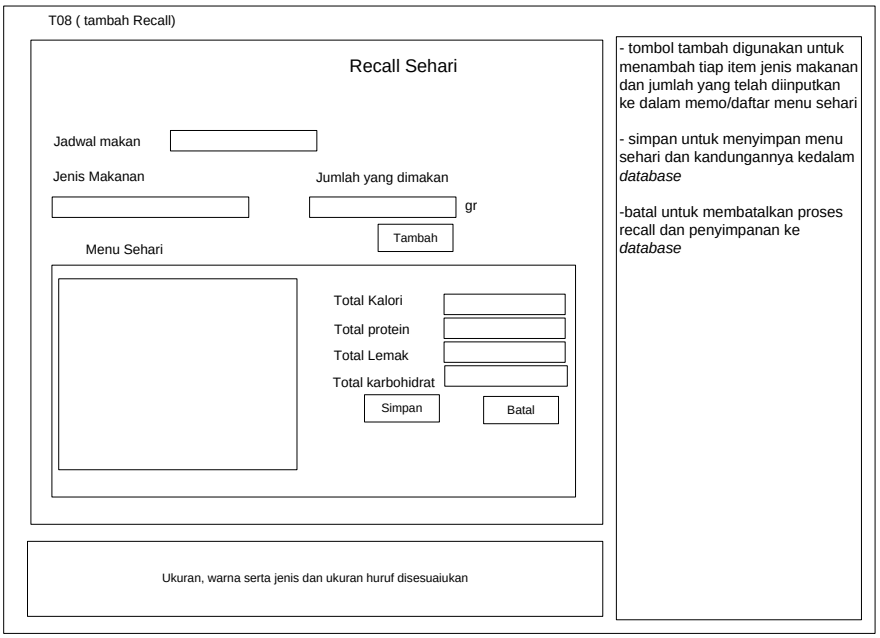
Gambar III.35 Perancangan antar muka proses anamnesa-Rancangan

7. Perancangan antar muka proses anamnesa-Kunjungan (T07)



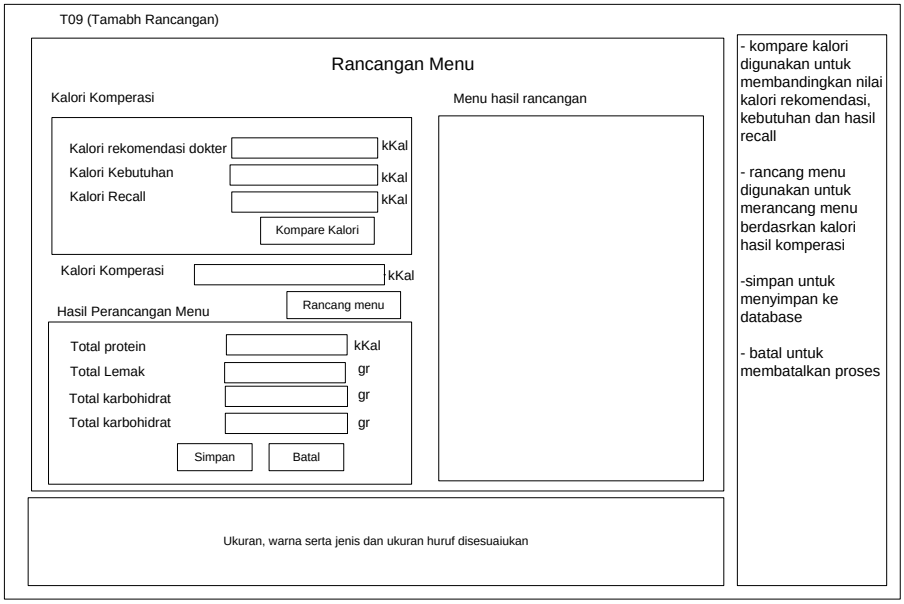
Gambar III.36 Perancangan antar muka proses anamnesa-Kunjungan

8. Perancangan antar muka recall sehari (T08)



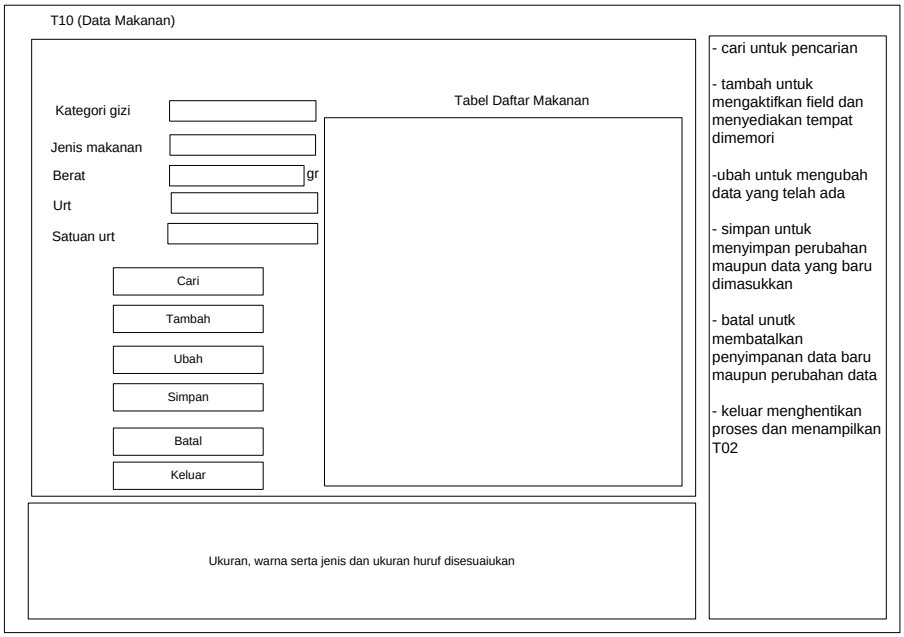
Gambar III.37 Perancangan antar muka recall sehari

9. Perancangan antar muka rancangan menu (T09)



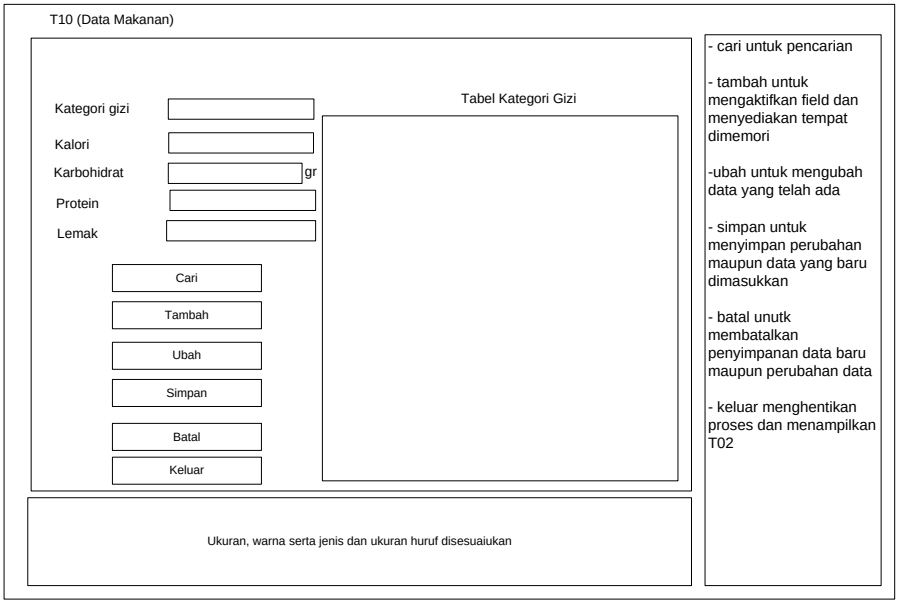
Gambar III.38 Perancangan antar muka rancangan menu

10. Perancangan antar muka data makanan (T10)



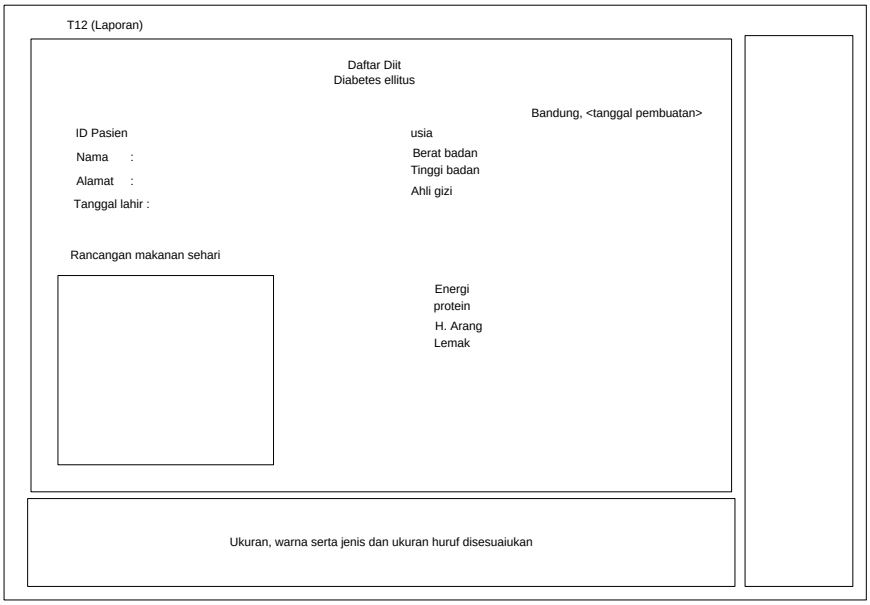
Gambar III.39 Perancangan antar muka data makanan

11. Perancangan antar muka data gizi (T11)



Gambar III.40 Perancangan antar muka data gizi

12. Perancangan antar muka laporan (T12)

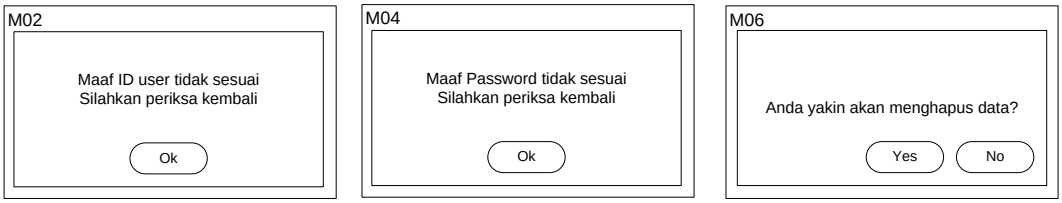


Gambar III.41 Perancangan antar muka laporan

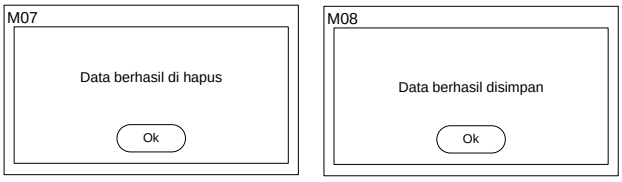
3.2.5.3 Perancangan antar muka pesan



Gambar III.42 Perancangan antar muka pesan 1

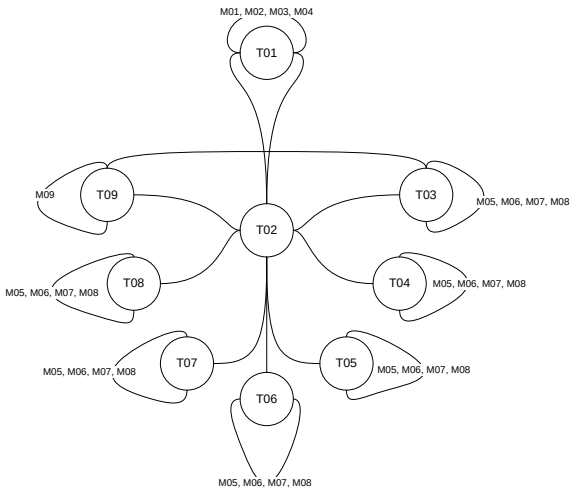


Gambar III.43 Perancangan antar muka pesan 2



Gambar III.44 Perancangan antar muka pesan 3

3.2.6 Jaringan simantik



Gambar III.45 Jaringan simantik