BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Sistem

Analisis sistem bertujuan untuk menguraikan permasalahan-permasalahan yang terdapat pada sistem dan menentukan kebutuhan-kebutuhan dari sistem yang akan dibangun. Analisis sistem meiputi analisis masalah, analisis sistem yang dibangun, analisis kebutuhan non fungsional, dan analisis kebutuhan fungsional.

Analisis sistem bertujuan untuk mengidentifikasi permasalahan-permasalahan yang ada pada sistem serta menentukan kebutuhan dari sistem yang dibangun. Analisis sistem tersebut meliputi analisis masalah, analisis sistem yang sedang berjalan, spesifikasi kebutuhan perangkat lunak dan analisis kebutuhan. Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponen dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, sehingga diperoleh solusi. Analisis sistem merupakan tahapan yang paling penting, karena kesalahan dalam tahap ini akan menyebabkan kesalahan di tahap selanjutnya.

3.1.1. Analisis Masalah

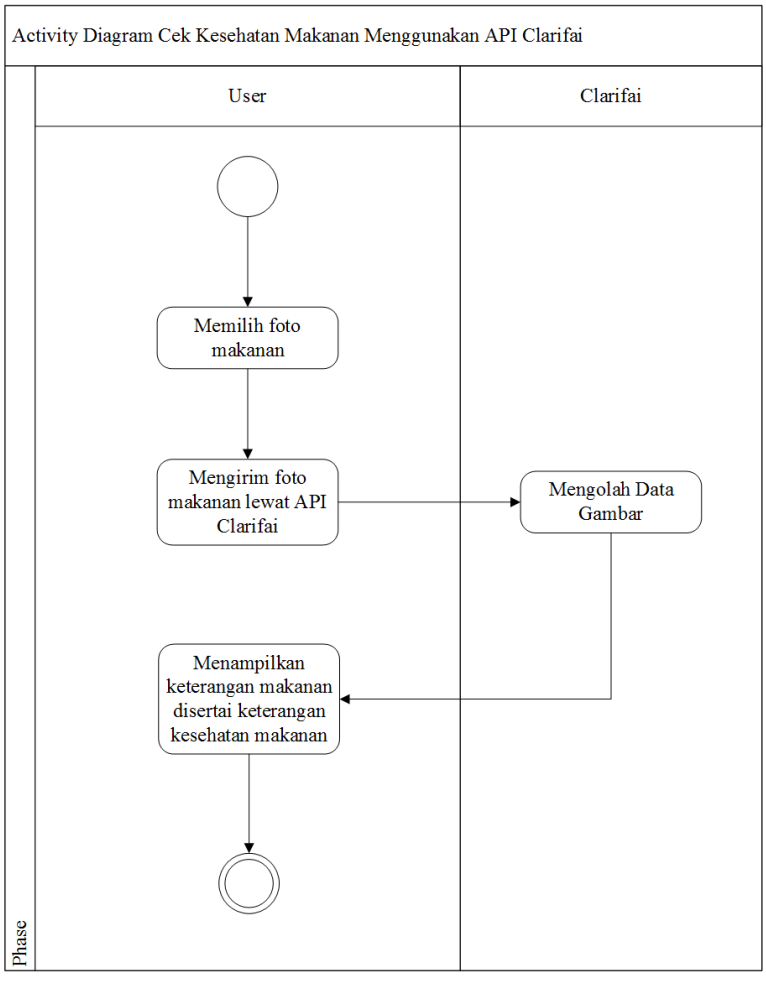
Karena belum adanya aplikasi pintar yang dapat memberikan rekomendasi makanan sehat. Maka pembangunan aplikasi untuk rekomendasi makanan sehat berbasis *mobile* merupakan media yang tepat, penyampaian informasi kepada *user* yang berupa informasi makanan sehat, serta informasi letak atau posisi tempat makan sehat yang ada di sekitar *user*.

Dan belum adanya aplikasi yang menyediakan informasi tempat makan yang menyediakan makanan sehat disekitar *user*, sehingga masyarakat tidak mengetahui tempat makan sehat mana yang berada disekitar *user*. Dengan membangun aplikasi yang menyediakan informasi secara realtime yang memberikan notifikasi ketika berada di tempat terdekat, maka mereka akan dapat menemukan tempat makan sehat yang sesuai dengan kebutuhan *user*.

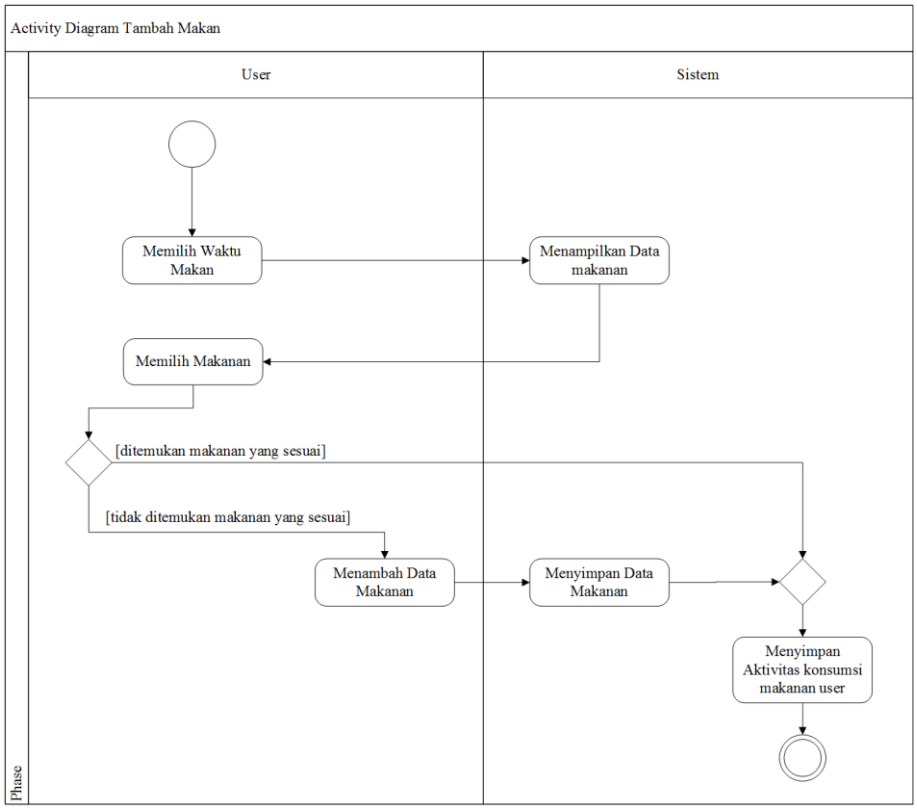
3.1.2. Analisis Sistem yang Dibangun

Analisis sistem yang dibangun merupakan gambaran lengkap sistem yang akan di bangun. Analisis sistem yang di bangun adalah sebagai berikut :

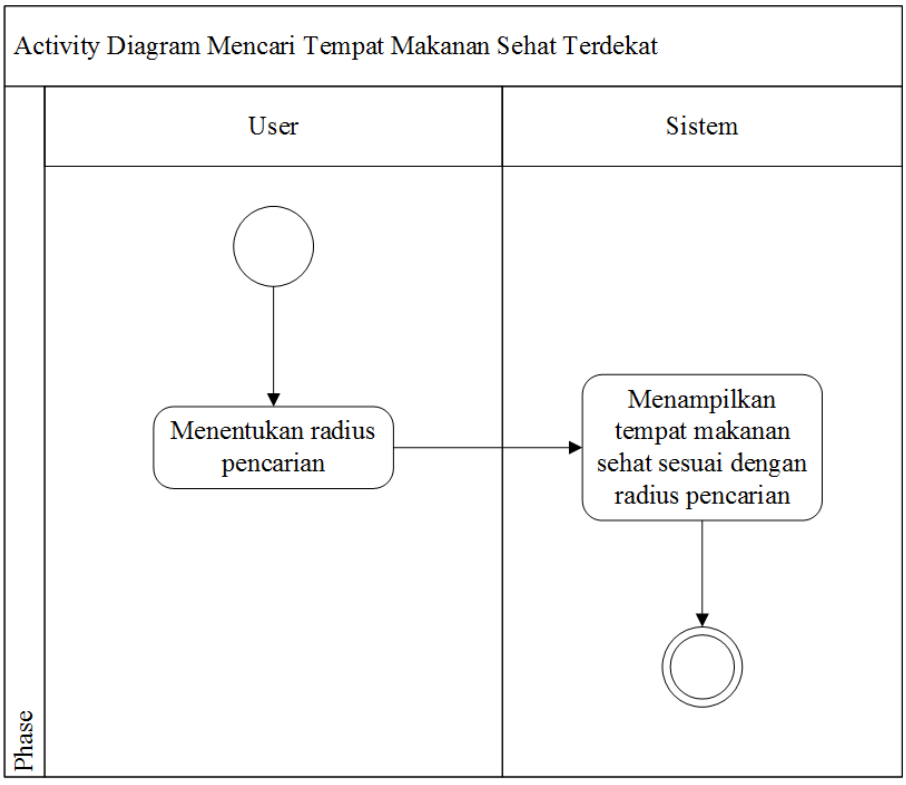
1. *User* memilih foto makanan
2. *User* mengirimkan foto makanan lewat API *Clarifai*
3. *Clarifai* mengolah data gambar
4. Sistem menampilkan keterangan makanan disertai keterangan kesehatan makanan



Gambar 3.1 Activity Diagram Cek Kedehatan Makanan Menggunakan API Clarifai



Gambar 3.2 Activity Diagram Tambah Makan

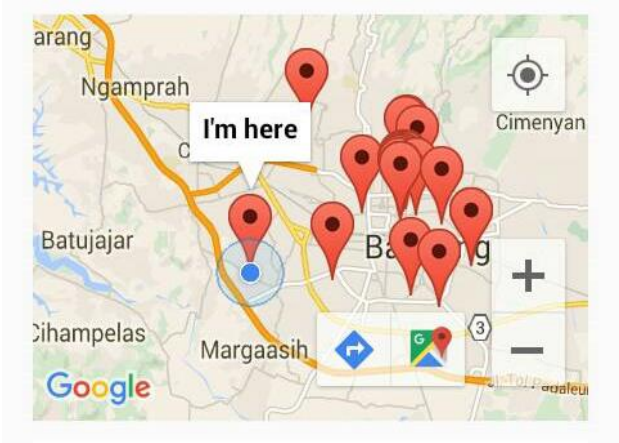


Gambar 3.3 Acitivity Diagram Mencari Tempat Makan Sehat Terdekat

3.1.3. Analisis Teknologi dan Metode

3.1.3.1. Analisis Teknologi *Google Maps*

Pada aplikasi ini untuk menunjukkan peta lokasi tempat makan menggunakan fitur *Google Maps*, seperti yang ditunjukkan pada Gambar 3.4 Peta Lokasi.



Gambar 3.4 Peta Lokasi

Fitur peta lokasi yang ada pada aplikasi ini diantaranya navigasi, petunjuk rute dan lokasi *user* berada.

a. Navigasi

Fitur ini memudahkan *user* mencapai berbagai tempat saat berkendara, berjalan kaki, atau bersepeda. Navigasi menginformasikan kondisi lalu lintas langsung, kapan harus berbelok, lajur mana yang digunakan, dan ketika ada rute yang lebih baik. Jika *user* keluar jalur, *Maps* akan otomatis menemukan rute baru.

b. Petunjuk Rute

Dengan fitur ini *user* dapat melihat daftar petunjuk arah langkah demi langkah. Baik menggunakan mobil, angkutan umum, berjalan kaki, atau sepeda. *User* juga dapat memilih di antara berbagai rute dan perkirakan waktu sampai pada lokasi tujuan.

c. Lokasi *User* Berada

Fitur ini menunjukkan titik lokasi dimana *user* sedang berada. Titik lokasi dapat berubah-ubah seiring dengan berpindah-pindahnya keberadaan *user*.

3.1.3.2. Analisis Metode *Haversine*

Metode *Haversine* digunakan untuk menghitung jarak antara titik di permukaan bumi menggunakan garis lintang (*longitude*) dan garis bujur (*latitude*) sebagai variabel *input*an. Dengan mengasumsikan bahwa bumi berbentuk bulat sempurna dengan jari-jari R 6.367, 45 km, dan lokasi dari 2 titik di koordinant bola (lintang dan bujur) masing-masing adalah lon1, lat1, dan lon2, lat2. Berikut Tabel 3.1 *latitude* dan *longitude* sebanyak 20 tempat makan yang telah di survei oleh peneliti. Penjelasan lengkap tabel data tempat makan terdapat pada bagian lampiran.

Tabel 3.1 Latitude dan Longitude Tempat Makan

Nama Tempat Makan	Latitude	Longitude
Warung Doeloer	-6.875675	107.577644
Warung Steak	-6.904462	107.612039
Pecel Lela	-6.904899	107.611399
Upnormal	-6.942809	107.591317
Bakmi Jowo	-6.903825	107.612634
Mie Merapi	-6.904899	107.611399
Nasi Goreng Mafia	-6.906680	107.613294
Kedai Lontong	-6.905004	107.611722
Bubur Ayam D U	-6.904378	107.613153
Pecel Lela	-6.90496	107.613119
Ayam Tulang Lunak Amboe	-6.903825	107.612634
Sindang Raso	-6.904240	107.612238
Bakso Aci Mas Jay	-6.90606	107.612261
Sate Sampang Madura	-6.910538	107.625986
Warung Frenzy	-6.939617	107.625109
Dcube Hangout Point	-6.907374	107.616435
Rumah Makan Banyumas	-6.9092706	107.614756
Steak Ranjang	-6.9141237	107.6155373
Verde Resto and	-6.906075	107.612904

lounge		
Mie Jakarta	-6.935372	107.614678

Sebagai contoh pada penelitian ini lokasi *user* sedang berada di kampus UNIKOM Jalan Dipatiukur Bandung dan ingin ke lokasi tempat Makan yang paling dekat dengan kampus UNIKOM.

Detail perhitungan jarak untuk semua tempat makan dapat dilihat pada bagian lampiran.

1. UNIKOM – Warung Doeloer

Lokasi 1 (Unikom): lat1= -6.886580, long1= 107.615263
Lokasi 2 (Warung Doeloer): lat2= -6.875675, long2= 107.577644

lat1 = -6.886580* 0.0174532925 radian = -0.120193495 radian
long1= 107.615263* 0.0174532925 radian = 1.878240663 radian
lat2 = -6.875675* 0.0174532925 radian = -0.120003167 radian
long2= 107.577644* 0.0174532925 radian= 1.877584087 radian
x = (long2-long1) * cos ((lat1+lat2)/2)
= (1.877584087-1.878240663) * cos ((-0.120193495+-0.120003167)/2)
= -0.000651846
y = (lat2-lat1)
= (-0.120003167- (-0.120193495))
= 0.000190328
d = sqrt (x*x + y*y) * R
= sqrt((-0.000651846*-0.000651846)+(0.000190328*0.000190328))*6371
= **4.326317239 km**

Berdasarkan hasil perhitungan menggunakan metode *haversine formula* diperoleh jarak dari UNIKOM ke Warung Doeloer adalah 4.326317239 km.

2. UNIKOM – Warung Steak

Lokasi 1 (Unikom): lat1= -6.886580, long1= 107.615263
Lokasi 2 (Warung Steak): lat2= -6.904462, long2= 107.612039

lat1 = -6.886580* 0.0174532925 radian = -0.120193495 radian

long1= 107.615263* 0.0174532925 radian = 1.878240663 radian

lat2 = -6.904462* 0.0174532925 radian = -0.120505595 radian

long2= 107.612039* 0.0174532925 radian= 1.878184393 radian

x = (long2-long1) * cos ((lat1+lat2)/2)

= (1.878184393-1.878240663) * cos ((-0.120193495+ -0.120505595)/2)

= -0.000055862

y = (lat2-lat1)

= (-0.120505595- (-0.120193495))

= -0.000312100

d = sqrt (x*x + y*y) * R

=sqrt((-0.000055862*-0.000055862)+(-0.000312100*-0.000312100))*6371

= **2.019987602 km**

Berdasarkan hasil perhitungan menggunakan metode *haversine formula* diperoleh jarak dari UNIKOM ke Warung Steak adalah 2.019987602 km.

3. UNIKOM – Pecel Lela

Lokasi 1 (Unikom): lat1= -6.886580, long1= 107.615263

Lokasi 2 (Pecel Lela): lat2= -6.904378, long2= 107.613153

lat1 = -6.886580* 0.0174532925 radian = -0.120193495 radian

long1= 107.615263* 0.0174532925 radian = 1.878240663 radian

lat2 = -6.904378* 0.0174532925 radian = -0.120504129 radian

long2= 107.613153* 0.0174532925 radian = 1.878203836 radian

x = (long2-long1) * cos ((lat1+lat2)/2)

= (1.878203836-1.878240663) * cos ((-0.120193495+(-0.120504129)/2)

= -0.000036560

y = (lat2-lat1)

= (-0.120504129-(-0.120193495))

= -0.000310634

d = sqrt (x*x + y*y) * R

=sqrt((-0.000036560*-0.000036560)+(-0.000310634* 0.000310634))*6371

= **1.992707185 km**

Berdasarkan hasil perhitungan menggunakan metode *haversine formula* diperoleh jarak dari UNIKOM ke Pecel Lela adalah 1.992707185 km.

4. UNIKOM – Upnormal

Lokasi 1 (Unikom): lat1= -6.886580, long1= 107.615263

Lokasi 2 (Upnormal): lat2= -6.904899, long2= 107.611399

lat1 = -6.886580* 0.0174532925 radian = -0.120193495 radian

long1= 107.615263* 0.0174532925 radian = 1.878240663 radian

lat2 = -6.904899* 0.0174532925 radian = -0.120513222 radian

long2= 107.611399* 0.0174532925 radian = 1.878173223 radian

x = (long2-long1) * cos ((lat1+lat2)/2)

= (1.878173223-1.878240663) * cos ((-0.120193495+(-0.120513222)/2)

= -0.000066952

y = (lat2-lat1)

= (-0.120513222-(-0.120193495))

= -0.000319727

d = sqrt (x*x + y*y) * R

=sqrt((-0.000066952*-0.000066952)+(-0.000319727*-0.000319727))*6371

= **2.081161008 km**

Berdasarkan hasil perhitungan menggunakan metode *haversine formula* diperoleh jarak dari UNIKOM ke Upnormal adalah 2.081161008 km.

5. UNIKOM – Bakmi Jowo

Lokasi 1 (Unikom): lat1= -6.886580, long1= 107.615263

Lokasi 2 (BakmI Jowo): lat2= -6.903825, long2= 107.612634

lat1 = -6.886580* 0.0174532925 radian = -0.120193495 radian

long1= 107.615263* 0.0174532925 radian = 1.878240663 radian

lat2 = -6.903825* 0.0174532925 radian = -0.120494477 radian

long2= 107.612634* 0.0174532925 radian = 1.878194778 radian

x = (long2-long1) * cos ((lat1+lat2)/2)

= (1.878194778-1.878240663) * cos ((-0.120193495+(-0.120494477)/2)

= -0.000045553

y = (lat2-lat1)

= (-0.120494477-(-0.120193495))

= -0.000300982

d = sqrt (x*x + y*y) * R

=sqrt((-0.000045553*-0.000045553)+(-0.000300982*-0.000300982))*6371

=**0.539393965 km**

Berdasarkan hasil perhitungan menggunakan metode *haversine formula* diperoleh jarak dari UNIKOM ke tempat makan Bakmi Jowo adalah 0.539393965 km.

Menurut hasil perhitungan yang telah dilakukan, maka tempat makan yang paling dekat dengan kampus UNIKOM adalah Bakmi Jowo. Sistem akan merekomendasikan Bakmi Jowo sebagai tempat makan terdekat dibandingkan dengan jarak tempat makan yang lain dengan *user* sesuai dengan hasil perhitungan yang telah dilakukan menggunakan metode *haversine formula*, dengan jarak 0.539393965 km.

3.1.3.3. Analisis Metode Pola Makan Sehat

Metode yang digunakan untuk menjaga pola makan sehat pada aplikasi ini adalah menghitung LMB (Laju Metabolisme Basal). LMB adalah jumlah kalori yang Anda bakar setiap hari hanya dengan bernafas. Untuk menghitungnya menggunakan rumus sebagai berikut :

a) Pria: (13,75 × berat badan) + (5 × tinggi badan) - (6,76 × usia) + 66
b) Wanita: (9,56 × berat badan) + (1,85 × tinggi badan) - (4,68 × usia) + 655

Contoh kasus :

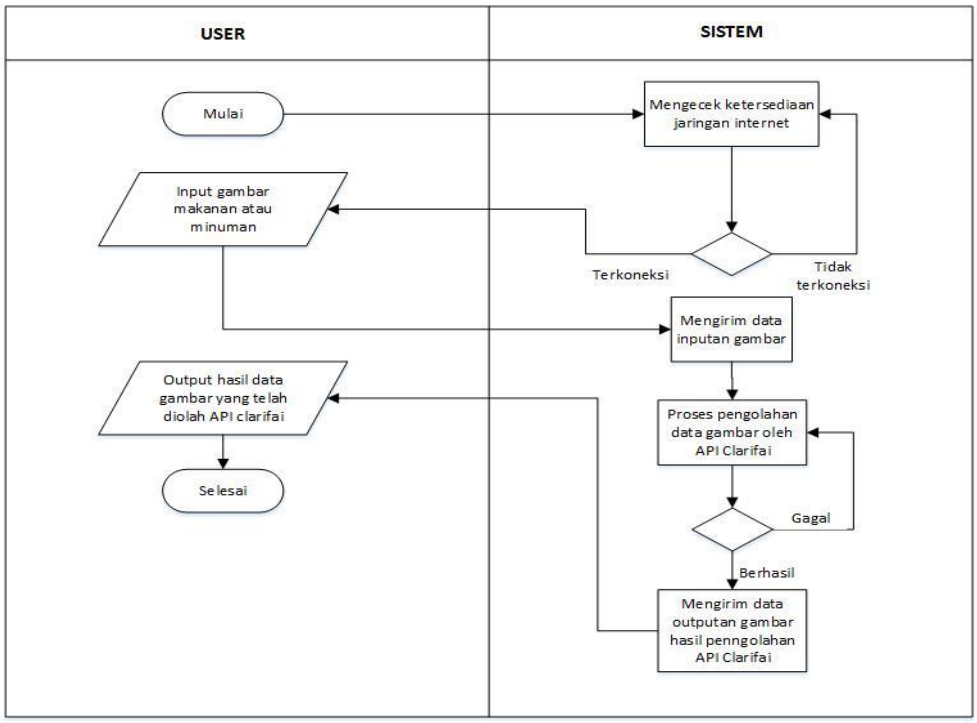
Seorang pria dengan usia 32 tahun, memiliki berat badan 60kg dan tinggi badan 170cm. maka didapatkan hasil sebagai berikut:

(13.75*60)+(5*170)-(6.76*32)+66 = 810+850-216.32+66= 15609.68

Jadi jumlah kalori yang dibutuhkan dalam 1 hari adalah 15609.68 atau dibulatkan menjadi 15609 kalori.

3.1.3.4. Analisis Data Masukan

Analisis data masukan bertujuan untuk mendefinisikan data masukan yang digunakan pada aplikasi ini. Berikut adalah Gambar 3.5 Diagram Alir Analisis Data Masukan pada API *Clarifai* :



Gambar 3.5 Diagram Alir Analisis Data Masukan pada API *Clarifai*

Penjelasan diagram alir penggunaan API *Clarifai* dapat dilihat dibawah ini :

1. Sistem memeriksa ketersediaan jaringan internet
2. *User* menginputkan gambar makanan atau minuman yang selanjutnya dikirim ke API *Clarifai*
3. Sistem Api *Clarifai* mengolah data gambar hasil inputan user
4. Sistem mengirim data hasil pengolahan gambar dari API *Clarifai*

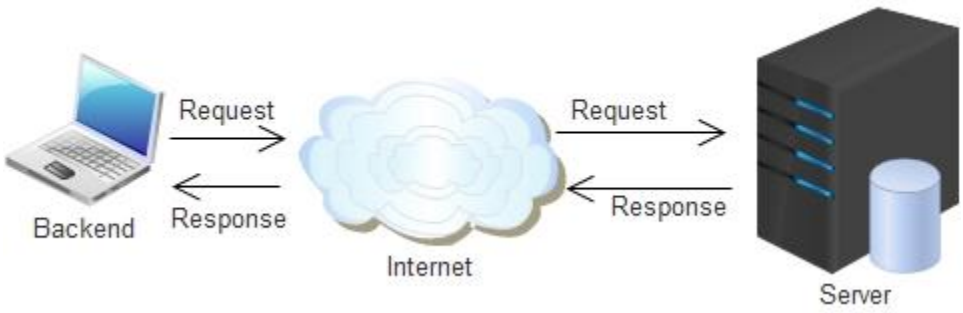
5. *User* menerima data gambar yang telah diolah API *Clarifai*

3.1.4. Analisis Arsitektur Sistem

Analisis merupakan tahapan awal yang sangat penting dalam membangun sebuah perangkat lunak *andorid*. Berikut adalah penjelasan dari masing-masing yang akan dibangun berdasarkan dua subsistem *web* dan *mobile*.

a. *Platform WEB*

Platform web adalah salah satu subsistem yang dipilih untuk pembangunan dari perangkat lunak ini. *User* perangkat lunak *platform* ini terbagi menjadi dua yaitu *admin*istrator dan *user*. *Admin*istrator bertugas untuk mengolah data konten *user*, Sedangkan untuk *user* digunakan untuk mengolah data catatan perjalanan *user*. Berikut adalah Gambar 3.6 Arsitektur sistem pada *Platform WEB*. Gambar ini menggambarkan secara keseluruhan arsitektur sistem pada *platform web*.



Gambar 3.6 Arsitektur sistem pada *platform web*

Berikut adalah deskripsi dari Gambar 3.6 Arsitektur sistem pada *Platform WEB* :

1. Sub sistem *web admin* dan *user* tempat makan melakukan *request* permintaan data ke *server* melalui jaringan *internet*
2. *Server* menerima *request* permintaan data dan mengambil data sesuai permintaan dari *database*.
3. *Server* mengirimkan data yang diminta melalui jaringan *internet* ke komputer *admin* dan *user* tempat makan.

b. *Platform Mobile*

Platform mobile adalah salah satu subsistem yang dipilih untuk pembangunan dari perangkat lunak ini. Arsitektur perangkat lunak pada *platform mobile* mengambarkan bagaimana perangkat lunak saling berinteraksi seperti diilustrasikan pada Gambar 3.7 Arsitektur sistem pada *platform andorid*. Gambar tersebut menggambarkan keseluruhan arsitektur sistem pada *platform mobile*.



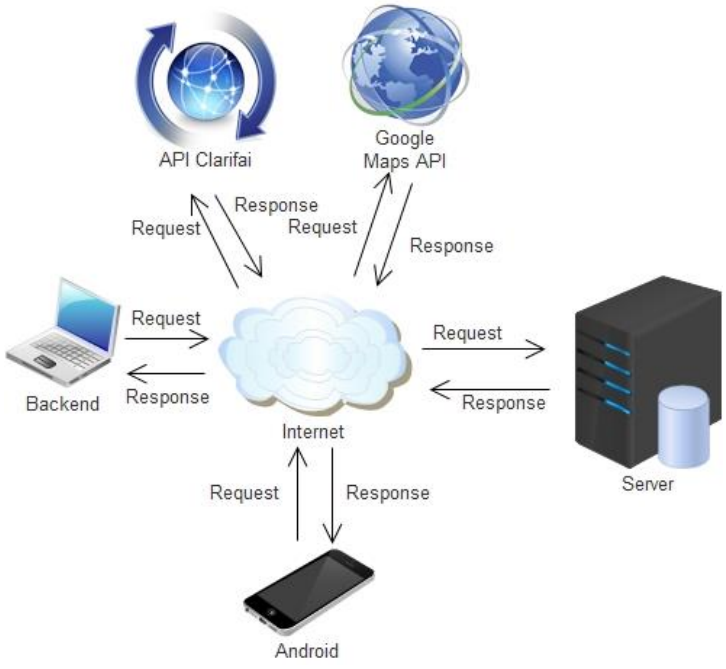
Gambar 3.7 Arsitektur sistem pada *platform andorid*

Berikut adalah deskripsi dari Gambar 3.7 Arsitektur sistem pada *platform andorid* :

1. Perangkat *mobile user* melakukan *request* data lokasi tempat makan sehat ke *server* melalui *Google Maps* API.
2. *Server* menerima *request* data dari *Google Maps* API dan menentukan jenis *request* yang diminta.
3. Jika *server* menerima permintaan lokasi tempat makan sehat, maka server akan langsung mengambil data yang ada di *database*.

4. Jika *server* menerima permintaan data *text* maka *server* akan langsung mengambil data yang ada di *database*.
5. Jika *server* menerima permintaan data gambar maka *server* akan langsung mengambil data yang ada di *database*.
6. Setelah *server* menerima data yang diminta data tersebut akan dikembalikan dalam bentuk JSON untuk diproses ke perangkat *mobile user*.
7. Perangkat *mobile user* melakukan *request* data kandungan kesehatan makanan dan minuman ke API *Clarifai*.
8. API *Clarifai* akan memproses permintaan data dari perangkat *mobile user* dan data akan dikirim kembali ke perangkat *mobile user* dalam bentuk gambar dan *text* kandungan kesehatan makanan dan minuman.

Arsitektur sistem yang digambarkan pada Gambar 3.7 Arsitektur sistem pada *Platform WEB* merupakan penggambaran sistem berdasarkan perspektif dua subsistem, sedangkan pada Gambar 3.8 Arsitektur sistem keseluruhan menggambarkan arsitektur sistem dari perspektif secara keseluruhan.



Gambar 3.8 Arsitektur Sistem Keseluruhan

3.1.5. Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak terbagi kedalam dua kebutuhan yaitu kebutuhan fungsional dan non fungsional. Kebutuhan fungsional pada perangkat lunak yang dibangun dapat dilihat pada Tabel 3.2 Kebutuhan Fungsional.

Tabel 3.2 Kebutuhan Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-F-001	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk registrasi kedalam sistem
SKPL-F-002	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk *login* kedalam sistem
SKPL-F-003	Subsistem *mobile* menyediakan fasilitas mereset *password login*
SKPL-F-004	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk memilih metode rekomendasi yang dibutuhkan
SKPL-F-005	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk mencari tempat makan sehat terdekat
SKPL-F-006	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk menambahkan data makanan dan minuman sehat
SKPL-F-007	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk meng*input* makanan atau minuman sehat
SKPL-F-008	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk melihat kandungan kesehatan makanan dan minuman
SKPL-F-009	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk melihat pencapaian kalori perhari
SKPL-F-010	Subsistem *mobile* menyediakan fasilitas bagi *user* untuk melihat history menu makanan atau minuman yang telah dikonsumsi perhari
SKPL-F-011	Subsistem *web* menyediakan fasilitas bagi *user* untuk daftar sebagai pemilik tempat makan sehat berdasarkan kepemilikannya
SKPL-F-012	Subsistem *web* menyediakan fasilitas *login* kedalam sistem untuk *user*
SKPL-F-013	Subsistem *web* menyediakan fasilitas *login* kedalam sistem untuk *admin*
SKPL-F-014	Subsistem *web* menyediakan fasilitas bagi *user* untuk menambahkan letak tempat makan sehat berdasarkan kepemilikannya
SKPL-F-015	Subsistem *web* menyediakan fasilitas bagi *admin*istrator untuk mengolah data menu makanan dan minuman sehat
SKPL-F-016	Subsistem *web* menyediakan fasilitas bagi *admin*istrator untuk mengolah

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
	data *user* yang terdaftar
SKPL-F-017	Subsistem *web* menyediakan fasilitas bagi *user* untuk menginvite *user* lain dalam penambahan letak tempat makan sehat berdasarkan kepemilikannya masing-masing

Kebutuhan fungsional dapat dilihat pada Tabel 3.3 Spesifikasi Kebutuhan Non Fungsional:

Tabel 3.3 Spesifikasi Kebutuhan Non Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-NF-001	Sistem yang dibangun memiliki dua subsistem yaitu *web* dan *mobile*
SKPL-NF-002	Sistem yang dibangun minimal menggunakan sistem operasi *andorid* versi 5.0 Lolipop
SKPL-NF-003	Sistem yang dibangun menggunakan perangkat kamera
SKPL-NF-004	Sistem dibangun dengan spesifikasi *hardware* yang memenuhi standar minimum kebutuhan
SKPL-NF-005	Desain *user interface* pada perangkat lunak yang akan dibangun mengacu kepada *design guidelines andorid* dari *google*
SKPL-NF-006	Sistem tidak mengizinkan konten yang mengandung unsur sara, Pornografi dan norma-norma kesusilaan.
SKPL-NF-007	Perangkat lunak yang dibangun harus menghasilkan format *file* standar (.json) yang bisa digunakan oleh pihak luar yang berkepentingan.

3.1.6. Analisis Kebutuhan Non Fungsional

Analisis Non Fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan aplikasi. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk aplikasi yang akan dibangun sampai dengan aplikasi tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan dalam sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

Pada analisis kebutuhan sistem non fungsional ini juga dijelaskan analisis mengenai perangkat keras (*hardware)*, perangkat lunak (*software)* dan *user* (*user)*

sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan aplikasi yang akan diterapkan.

Analisis kebutuhan non fungsional pada sistem ini meliputi analisis perangkat lunak, analisis perangkat keras dan perangkat pikir.

1. Product Requirement

   Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *product requirement*.

   a. SKPL-NF-001 Sistem yang dibangun memiliki dua subsistem yaitu *web* dan *mobile*.
   b. SKPL-NF-002 Sistem yang dibangun minimal menggunakan sistem operasi *andorid* versi 5.0 Lolipop dan dibangun dengan dengan kebutuhan perangkat lunak minimum :

Tabel 3.4 Kebutuhan Perangkat Lunak

No	Kebutuhan Perangkat Lunak
1	Sistem Operasi (*platform* bebas)
2	*Web server* XAMPP versi 1.1.7 ke atas sebagai *web server*
3	Kode editor berupa sublime *text* 3
4	Integrated development environment (IDE) berupa *Andorid* Studio
5	*Andorid* Development Tools versi 22.1.6 sebagai plugins *Andorid* Studio
6	*Andorid Software* Development Kit sebagai framework *andorid*
7	*Text* editor.
8	*Web* Browser

   c. SKPL-NF-003 Sistem yang dibangun menggunakan perangkat kamera
   d. SKPL-NF-004 Sistem dibangun dengan spesifikasi *hardware* yang memenuhi standar minimum kebutuhan seperti:

Tabel 3.5 Kebutuhan Perangkat Keras *Web*

No	Kebutuhan Perangkat Keras
1	RAM 512 MB
2	Hardisk 180 GB
3	Koneksi *Internet*
4	Mouse, Keyboard, Monitor

Tabel 3.6 Kebutuhan Perangkat Keras *Mobile*

No	Kebutuhan Perangkat *Mobile*
1	Sistem Operasi minimal Lollipop versi 5.0
2	*Random Access Memory* (RAM) 1 GB
3	Memiliki fitur *Global Positioning* Sistem (GPS).
4	Memiliki Jaringan *Internet* 3G atau WLAN Wifi

2. Organisational Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *Organisational Requirement*.

a. SKPL-NF-005 Desain *user interface* pada perangkat lunak yang akan dibangun mengacu kepada *design guidelines andorid* dari *google*

3. External Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *External Requirement.*

a. SKPL-NF-006 Sistem tidak mengizinkan konten yang mengandung unsur sara, pornografi dan norma-norma kesusilaan.

b. SKPL-NF-007 Perangkat lunak yang dibangun harus menghasilkan format *file* standar (.json) yang bisa digunakan oleh pihak luar yang berkepentingan.

4. Analisis Kebutuhan Non Fungsional Perangkat Pikir

Analisis kebutuhan non fungsional perangkat pikir adalah batasan-batasan dari layanan-layanan dan fungsi-fungsi yang dibutuhkan dari sebuah sistem dari segi perangkat pikir. Dalam penelitian ini dikelompokkan menjadi 3 kategori yaitu *User Knowledge and Experience, User Job and Task Characteristic and User Physical Characteristic.*

a. *User Knowledge and Experience*

User Knowledge and Experience yang diharapkan dari *user* perangkat lunak pada penelitian ini dapat dilihat pada Tabel 3.7 *User Knowledge and Experience.*

Tabel 3.7 *User Knowledge and Experience*

Knowledge And Experiences

Knowledge And Experiences	
Komputer Literacy	Cukup sampai tinggi
Sistem *Experiences*	Rendah sampai tinggi
Application Experience	Rendah sampai tinggi
Task Experience	Rendah sampai tinggi
Other Sistem *Use*	Frequent
Education	SMA sampai Perguruan Tinggi
Reading Level	Sedang sampai Tinggi
Typing Skill	Sedang sampai Tinggi
Native Language Or Culture	English

b. *User Job and Task Characteristic*

User Job and Task Characteristic yang diharapkan dari *user* perangkat lunak pada penelitian ini dapat dilihat pada Tabel 3.8 *Job and Task Characteristic*

Tabel 3.8 *Job and Task Characteristic*

Job and Task Characteristic	
Task Structure	Tinggi
Social Interactions	Perlu
Primary Training	None
Job Category	*Traveler*
Frequency of Use	Rendah
Task or Need Importance	*Medium*

c. *User* Physical Characteristic

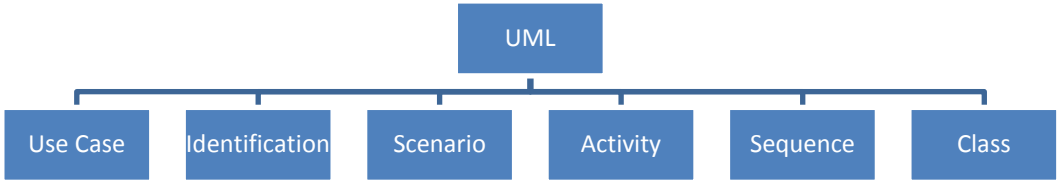
User Physical Characteristic yang diharapkan dari *user* perangkat lunak pada penelitian ini dapat dilihat pada Tabel 3.9 *User Physical Characteristic.*

Tabel 3.9 User *Physical Characteristic*

User Physical Characteristic	
Age	*Young, middle aged, and elderly.*
Gender	*Male and Female*
Handedness	*Both*
Disabilities	*None*

3.1.7. Analisis Kebutuhan Fungsional *Mobile*

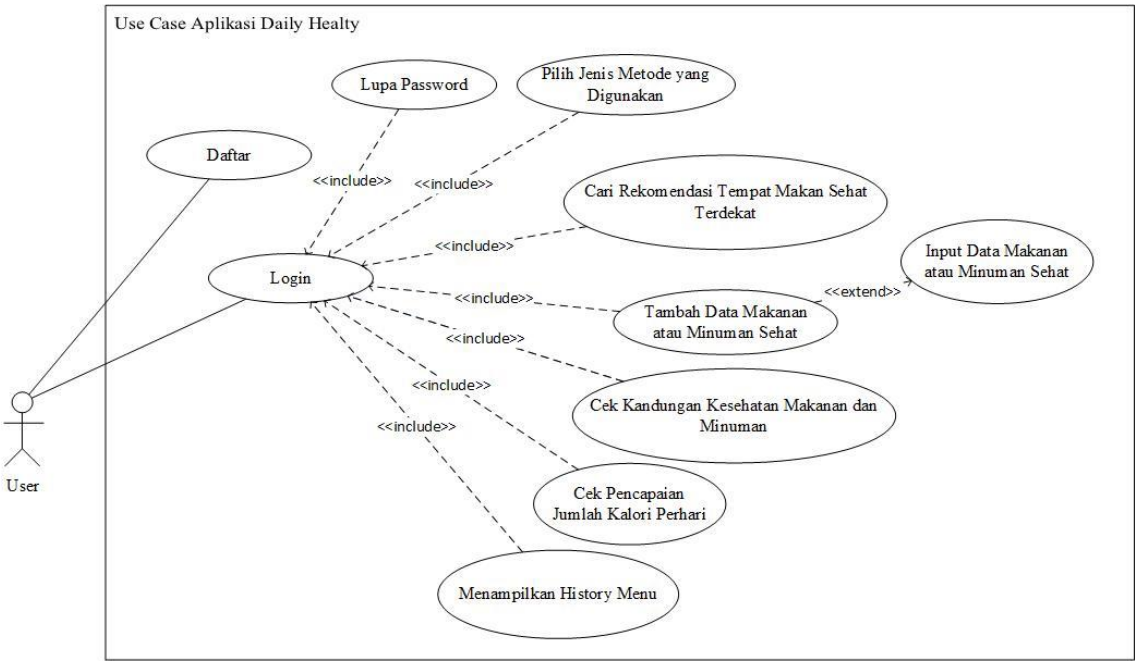
Analisis kebutuhan fungsional bertujuan untuk menganalisis proses yang diterapkan dalam sistem dan menjelaskan kebutuhan yang diperlukan. Menganalisis operasional sistem dengan mendefinisikan skenario penggunaan aplikasi.



Gambar 3.9 Fungsional *Diagram*

3.1.7.1. *Use Case Diagram*

Use Case adalah konstruksi untuk mendeskripsikan bagaimana sistem terlihat di mata *user*. Sasaran pemodelan *Use Case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan.



Gambar 3.10 *Use Case Diagram*

3.1.7.2. *Use Case Identification*

Tabel 3.10 ***Use Case Identification User***

Use Case Aplikasi *Andorid*		
No	*Use Case*	Deskripsi
SKPL-U-01	Daftar	Sistem menampilkan form daftar untuk *user* mendaftar sebagai member.
SKPL-U-02	*Login*	Sistem menampilkan form *login* untuk *user* yang terdiri dari *email* dan *password*
SKPL-U-03	Lupa *Password*	Sistem menampilkan form lupa *password* yang terdiri dari *email*
SKPL-U-04	Pilih Jenis Metode yang Digunakan	Sistem menampilkan menu pilihan metode yang direkomendasikan oleh sistem
SKPL-U-05	Cari Rekomendasi Tempat Makan Terdekat	Sistem menampilkan menu pencarian rekomendasi tempat makan sehat yang berada di sekitar *user*
SKPL-U-06	Tambah Data Makanan atau Minuman Sehat	Sistem menampilkan form tambah makan atau minuman sesuai kebutuhan *user*
SKPL-U-07	*Input* Data Makanan atau Minuman Sehat	Sistem menampilkan form *input*an makan atau minuman sesuai kebutuhan *user*
SKPL-U-08	Cek Kandungan Kesehatan Makanan dan Minuman	Sistem menampilkan menu cek kandungan kesehatan makanan dan minuman menggunakan teknologi kamera pada *andorid*
SKPL-U-09	Cek Pencapaian Jumlah Kalori Perhari	Sistem menampilkan menu cek pencapaian jumlah kalori perhari yang *user* konsumsi
SKPL-U-10	Menampilkan History Menu	Sistem menampilkan menu history menu makanan atau minuman yang telah *user* konsumsi

3.1.7.3. *Use Case Scenario*

Skenario *Use Case* berfungsi untuk menjelaskan lebih detail mengenai modul-modul yang terdapat dalam aplikasi. Adapun penjelasan mengenai keterangan tabel skenario *Use Case* berdasarkan aktor sebagai berikut :

1. *Use Case Scenario* Daftar

   Use Case scenario daftar menggambarkan langkah-langkah aktor untuk mendaftar kedalam sistem. *Use Case scenario* daftar dapat dilihat pada Tabel 3.11 *Use Case* Scenario Daftar dibawah ini.

Tabel 3.11 *Use Case Scenario* Daftar

Use Case Name	Daftar	
Related Requirements	SKPL-F-001	
Goal Con*text*	*User* berhasil mendaftar data pada sistem	
Preconditions	1. *User* belum *login* kedalam system 2. Sistem belum menampilkan form daftar	
Successful End Condition	Sistem berhasil mendaftar di sistem	
Failed End Condition	Sistem gagal menyimpan data daftar *user* di sistem	
Primary Actor	*User*	
Trigger	*User* menekan menu daftar	
Main Flow	Step	Action
	1	Menekan tombol daftar
	2	Memproses form pendaftaran
	3	Menampilkan form pengisian daftar
	4	Mengisi form daftar
	5	Menekan tombol simpan
	6	Memvalidasi form daftar
	7	Menampilkan pesan data *user* pendaftar berhasil disimpan di simpan
Extension	Step	Branching Action
	3.1	Gagal menampilkan form daftar
	7.1	Menampilkan pesan kesalahan dalam daftar

2. *Use Case Scenario Login*

Use Case scenario login menggambarkan langkah-langkah aktor untuk masuk kedalam sistem. *Use Case scenario login* dapat dilihat pada Tabel 3.12 *Use Case* Scenario *Login* dibawah ini.

Tabel 3.12 *Use Case Scenario Login*

Use Case Name	*Login*	
Related Requirements	SKPL-F-002	
Goal Con*text*	*User* berhasil *login* kedalam sistem	
Preconditions	1. *User* belum *login* kedalam sistem 2. Sistem belum menampilkan form *login*	
Successful End Condition	Sistem berhasil menampilkan form *login*	
Failed End Condition	Sistem gagal menampilkan form *login*	
Primary Actor	*User*	
Trigger	*User* menekan *login*	
Main Flow	Step	Action
	1	Menekan tombol *login*
	2	Memproses form *login*
	3	Menampilkan form *login*
	4	Mengisi form *login*
	5	Menekan tombol simpan
	6	Memvalidasi form *login*
	7	Menampilkan pesan berhasi *login* kedalam aplikasi
Extension	Step	Branching Action
	3.1	Gagal menampilkan form *login*
	7.1	Menampilkan pesan gagal *login*

3. *Use Case Scenario* Lupa *Password*

Use Case scenario Lupa *Password* menggambarkan langkah-langkah aktor untuk mengubah *password* apabila aktor lupa *password*. *Use Case scenario* lupa *password* dapat dilihat pada Tabel 3.13 *Use Case* Scenario Lupa *Password* dibawah ini.

Tabel 3.13 *Use Case Scenario* Lupa *Password*

Use Case Name	Lupa *Password*
Related Requirements	SKPL-F-003

Goal Con*text*	Sistem mengirim *email* kepada *user*	
Preconditions	1. *User* belum *login* kedalam sistem 2. *User* belum meng*input*kan *email*	
Successful End Condition	Sistem berhasil mengirim *email* kepada *user*	
Failed End Condition	Sistem gagal mengirim *email* kepada *user*	
Primary Actor	*User*	
Trigger	*User* menekan menu lupa *password*	
Main Flow	Step	Action
	1	Masuk menu lupa *password*
	2	Memproses form lupa *password*
	3	Menampilkan form lupa *password*
	4	Mengisi form lupa *password*
	5	Menekan tombol kirim
	6	Memvalidasi form lupa *password*
	7	Menampilkan pesan *password* telah terkirim di *email user*
Extension	Step	Branching Action
	3.1	Gagal menampilkan form lupa *password*
	7.1	Menampilkan pesan tidak valid

4. *Use Case Scenario* Pilih Jenis Metode yang Digunakan

Use Case scenario pilih jenis metode yang digunakan menggambarkan langkah-langkah aktor untuk memilih metode yang ingin digunakan aktor. *Use Case scenario* pilih jenis metode yang digunakan dapat dilihat pada Tabel 3.14 *Use Case* Scenario Pilih Jenis Metode yang Digunakan dibawah ini.

Tabel 3.14 *Use Case Scenario* Pilih Jenis Metode yang Digunakan

Use Case Name	Pilih metode yang digunakan
Related Requirements	SKPL-F-004
Goal Con*text*	*User* berhasil memilih metode yang digunakan
Preconditions	1. *User* berhasil *login* kedalam system 2. Sistem belum menampilkan pilihan metode yang digunakan
Successful End Condition	Sistem berhasil menyimpan metode yang digunakan
Failed End Condition	Sistem gagal menyimpan metode yang digunakan

Primary Actor	*User*	
Trigger	*User* menekan menu pilih metode	
Main Flow	Step	Action
	1	Masuk menu pilihan metode rekomendasi
	2	Memproses menu pilihan metode rekomendasi
	3	Menampilkan pilihan metode rekomendasi
	4	Memilih salah satu menu pilihan metode rekomendasi
	5	Memproses metode pilihan *user*
	6	Masuk menu utama
Extension	Step	Branching Action
	3.1	Gagal menampilkan pilihan metode rekomendasi
	6.1	Gagal masuk menu utama aplikasi

5. *Use Case Scenario* Cari Rekomendasi Tempat Makan Terdekat

Use Case scenario cari rekomendasi tempat makan terdekat menggambarkan langkah-langkah aktor untuk mencari tempat makan sehat terdekat dengan aktor. *Use Case scenario* cari rekomendasi tempat makan terdekat dapat dilihat pada Tabel 3.15 *Use Case* Scenario Cari Rekomendasi Tempat Makan Terdekat dibawah ini.

Tabel 3.15 *Use Case Scenario* Cari Rekomendasi Tempat Makan Terdekat

Use Case Name	Cari rekomendasi tempat makan terdekat	
Related Requirements	SKPL-F-005	
Goal Con*text*	*User* berhasil mendapatkan letak rekomendasi tempat makan sehat	
Preconditions	*User* berhasil *login* / sudah *login* kedalam sistem	
Successful End Condition	Sistem berhasil mendapatkan letak rekomendasi tempat makan sehat	
Failed End Condition	Sistem gagal mendapatkan letak rekomendasi tempat makan sehat	
Primary Actor	*User*	
Trigger	*User* menekan menu cari tempat makan sehat	
Main Flow	Step	Action
	1	Menekan tombol cari tempat makan sehat
	2	Memproses pencarian rekomendasi tempat makan sehat sekitar
	3	Menampilkan hasil pencarian rekomendasi tempat makan sehat

	4	Memilih salah satu rekomendasi tempat makan sehat sekitar
	5	Memproses pilihan rekomendasi tempat makan sehat
	6	Menampilkan petunjuk arah rekomendasi tempat makan sehat
Extension	Step	Branching Action
	3.1	Gagal menampilkan hasil pencarian rekomendasi tempat makan sehat sekitar
	5.1	Gagal menampilkan petunjuk arah rekomendasi tempat makan sekitar

6. *Use Case Scenario* Tambah Data Makanan atau Minuman

Use Case scenario tambah data makanan atau minuman menggambarkan langkah-langkah aktor untuk menambah sendiri data makanan atau minuman sehat. *Use Case scenario* tambah data makanan atau minuman dapat dilihat pada Tabel 3.16 *Use Case* Scenario Tambah Data Makanan atau Minuman dibawah ini.

Tabel 3.16 *Use Case Scenario* Tambah Data Makanan atau Minuman

Use Case Name	Tambah data makanan atau minuman sehat	
Related Requirements	SKPL-F-006	
Goal Con*text*	*User* berhasil menyimpan data makanan atau minuman sehat	
Preconditions	*User* berhasil *login* / sudah *login* kedalam sistem	
Successful End Condition	Sistem berhasil menyimpan data makanan dan minuman sehat	
Failed End Condition	Sistem gagal menyimpan data makanan dan minuman sehat	
Primary Actor	*User*	
Trigger	*User* menekan menu tambah menu makanan atau minuman	
Main Flow	Step	Action
	1	Menekan tombol tambah makanan atau minuman
	2	Memproses form tambah makanan atau minuman
	3	Menampilkan form tambah makanan atau minuman
Extension	Step	Branching Action
	3.1	Gagal menampilkan form tambah makanan atau minuman

7. *Use Case Scenario Input* Data Makanan atau Minuman

Use Case scenario input data makanan atau minuman menggambarkan langkah-langkah aktor menambahkan data makanan atau minuman sehat. *Use Case scenario input* data makanan atau minuman dapat dilihat pada Tabel 3.17 *Use Case* Scenario *Input* Data Makanan atau Minuman dibawah ini.

Tabel 3.17 *Use Case Scenario* Input Data Makanan atau Minuman

Use Case Name	*Input* data makanan atau minuman	
Related Requirements	-	
Goal Con*text*	*User* berhasil menyimpan data makanan atau minuman	
Preconditions	1. *User* berhasil *login* / sudah *login* kedalam sistem 2. *User* sudah masuk menu tambah data makanan atau minuman sehat	
Successful End Condition	Sistem berhasil menyimpan data makanan dan minuman sehat	
Failed End Condition	Sistem gagal menyimpan data makanan atau minuman	
Primary Actor	*User*	
Trigger	*User* meng*input* data makanan atau minuman	
Main Flow	Step	Action
	1	Mengisi form *input* tambah makanan atau minuman sehat
	2	Menekan tombol simpan
	3	Memvalidasi *input*an data makanan atau minuman sehat
	4	Berhasil menyimpan data makanan atau minuman sehat
Extension	Step	Branching Action
	3.1	Gagal menyimpan data makanan atau minuman sehat

8. *Use Case Scenario* Cek Kandungan Kesehatan Makanan dan Minuman

Use Case scenario cek kandungan kesehatan makanan dan minuman menggambarkan langkah-langkah aktor mengecek kandungan kesehatan makanan atau minuman sehat. *Use Case scenario* cek kandungan kesehatan makanan dan minuman dapat dilihat pada Tabel 3.18 *Use Case* Scenario *Input* Data Makanan atau Minuman dibawah ini.

Tabel 3.18 *Use Case Scenario* Cek Kandungan Kesehatan Makanan dan Minuman

Use Case Name	Cek kandungan kesehatan makanan atau minuman
Related Requirements	SKPL-F-007

Goal Con*text*	*User* berhasil mendapatkan informasi kandungan kesehatan makanan dan minuman	
Preconditions	1. *User* berhasil *login* / sudah *login* kedalam sistem 2. Kandungan kesehatan dan minuman akan diperiksa menggunakan API *Clarifai* 3. Perangkat *andorid user* harus mempunyai teknologi camera	
Successful End Condition	Sistem berhasil mendapatkan informasi kandungan kesehatan makanan dan minuman	
Failed End Condition	Sistem gagal mendapatkan kandungan kesehatan dan minuman	
Primary Actor	*User*	
Trigger	*User* menekan tombol cek kandungan kesehatan	
Main Flow	Step	Action
	1	Menekan tombol cek kandungan kesehatan makanan atau minuman
	2	Memproses fitur cek kandungan sesehatan makanan atau miunman menggunakan teknologi kamera pada *andorid*
	3	Menampilkan fitur cek kandungan kesehatan makanan atu minuman menggunakan teknologi kamera pada *andorid*
	4	Mentag gambar makanan atau minuman
	5	Menekan tombol cek
	6	Memvalidasi data gambar yang telah di tag
	7	Data tag gambar berhasil dikirim ke *server* API *clarifai*
	8	Data tag gambar di proses sistem API *Clarifai*
	9	*Server* API *clarifai* mengirimkan hasil cek kandungan kesehatan makanan atau minuman
	10	*User* menerima data kandungan makanan atau minuman yang telah di periksa kandungan kesehatannya
Extension	Step	Branching Action
	3.1	Gagal menampilkan fitur cek kandungan kesehatan makanan atau minuman
	7.1	Data tag gambar tidak valid

9. *Use Case Scenario* Cek Pencapaian Jumlah Kalori Perhari

Use Case scenario cek pencapaian jumlah kalori perhari menggambarkan langkah-langkah aktor mengecek pencapain jumlah kalori perhari. *Use Case*

scenario cek pencapaian jumlah kalori perhari dapat dilihat pada Tabel 3.19 *Use Case* Scenario cek pencapaian jumlah kalori perhari dibawah ini.

Tabel 3.19 *Use Case scenario* cek pencapaian jumlah kalori perhari

Use Case Name	Cek pencapaian jumlah kalori perhari	
Related Requirements	SKPL-F-008	
Goal Con*text*	*User* berhasil mendapatkan data pencapaian jumlah kalori perhari	
Preconditions	*User* berhasil *login* / sudah *login* kedalam sistem	
Successful End Condition	Sistem berhasil menampilkan data pencapaian jumlah kalori perhari	
Failed End Condition	Sistem gagal menampilkan data pencapaian jumlah kalori perhari	
Primary Actor	*User*	
Trigger	*User* menekan menu cek kalori	
Main Flow	Step	Action
	1	Menekan tombol cek pencapaian kalori
	2	Memproses data pencapaian kalori perhari
	3	Menampilkan data keseluruhan pencapaian kalori perhari
Extension	Step	Branching Action
	3.1	Gagal memproses data pencapaian kalori perhari

10. *Use Case Scenario* Menampilkan *History* Menu

Use Case scenario menampilkan *history* menu menggambarkan langkah-langkah aktor untuk menampilkan history menu perhari. *Use Case scenario* menampilkan *history* menu dapat dilihat pada Tabel 3.20 *Use Case* scenario menampilkan history menu dibawah ini.

Tabel 3.20 *Use Case scenario* menampilkan history menu

Use Case Name	Menampilkan history menu	
Related Requirements	SKPL-F-009	
Goal Con*text*	*User* berhasil mendapatkan informasi history menu	
Preconditions	*User* berhasil *login* / sudah *login* kedalam sistem	
Successful End Condition	Sistem berhasil menampilkan history menu	
Failed End Condition	Sistem gagal menampilkan history menu	
Primary Actor	*User*	
Trigger	*User* menekan menu history menu	
Main Flow	Step	Action

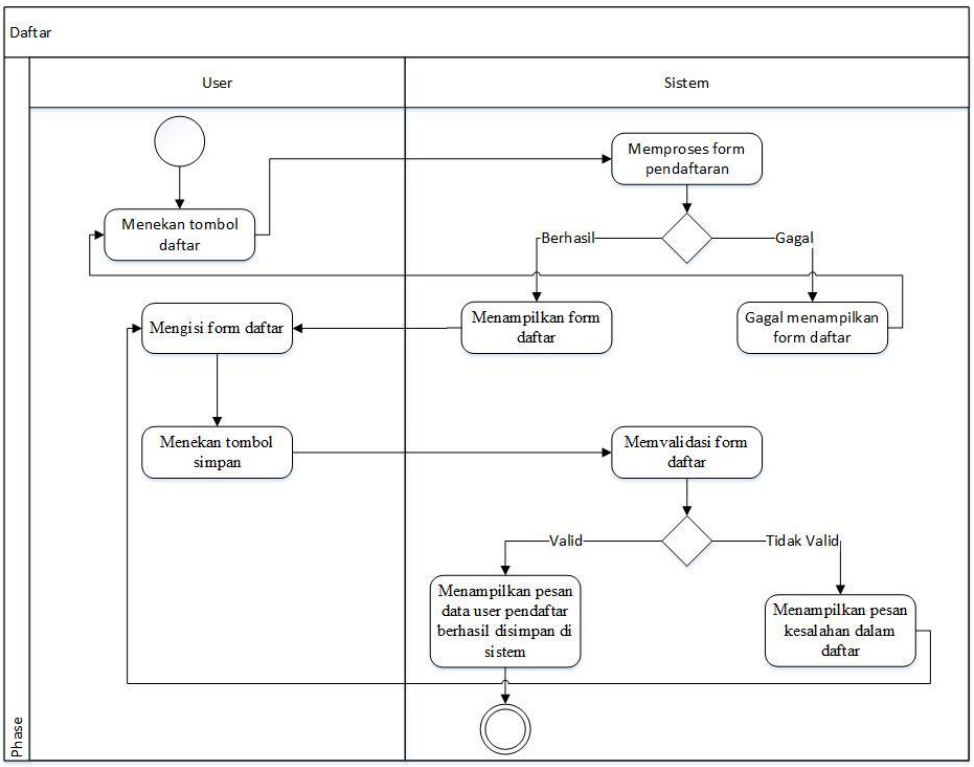
	1	Menekan tombol history menu
	2	Memproses data history menu
	3	Menampilkan data history menu
Extension	Step	Branching Action
	3.1	Gagal menampilkan data history menu

3.1.7.4. *Activity Diagram*

Diagram aktifitas menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir. Penggambaran *activity diagram* memodelkan *event-event* yang terjadi pada *Use Case* dan digunakan untuk pemodelan aspek dinamis dari aplikasi. Diagram aktifitas dimodelkan seperti gambar sebagai berikut:

a) *Activity* Diagram Daftar

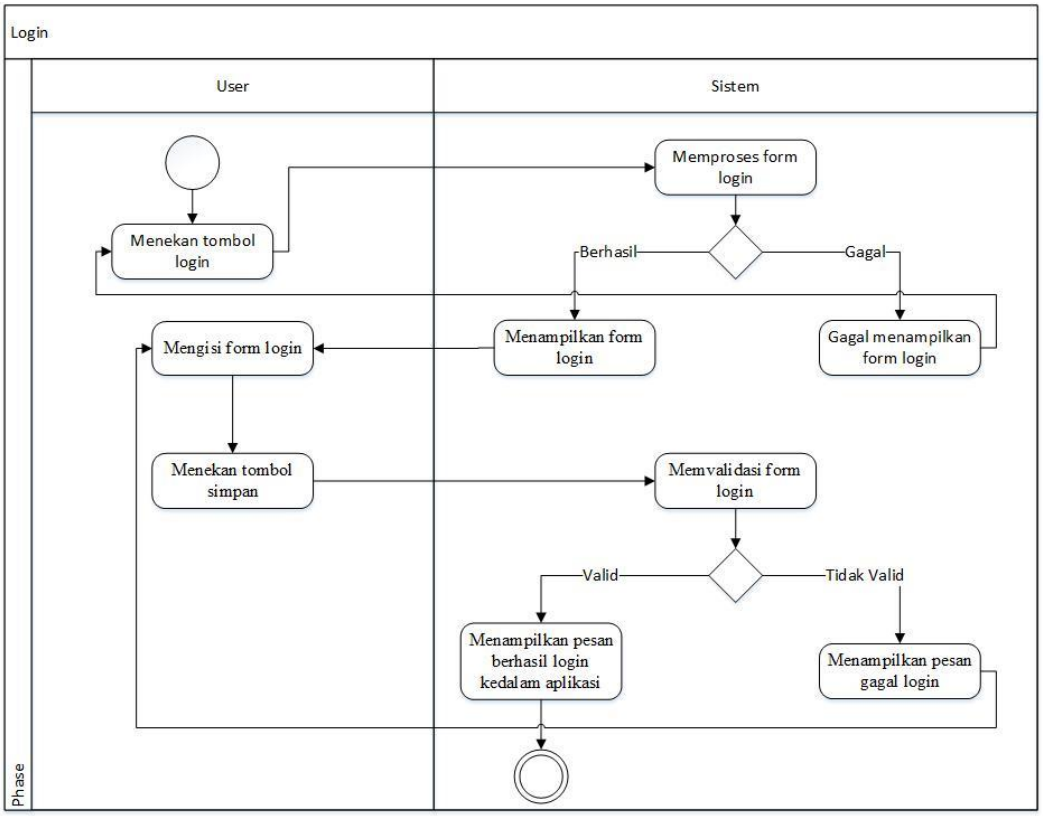
Berikut adalah Gambar 3.11 *Activity* Diagram Daftar yang menggambarkan *activity* diagram daftar :



Gambar 3.11 *Activity Diagram* Daftar

b) *Activity* Diagram *Login*

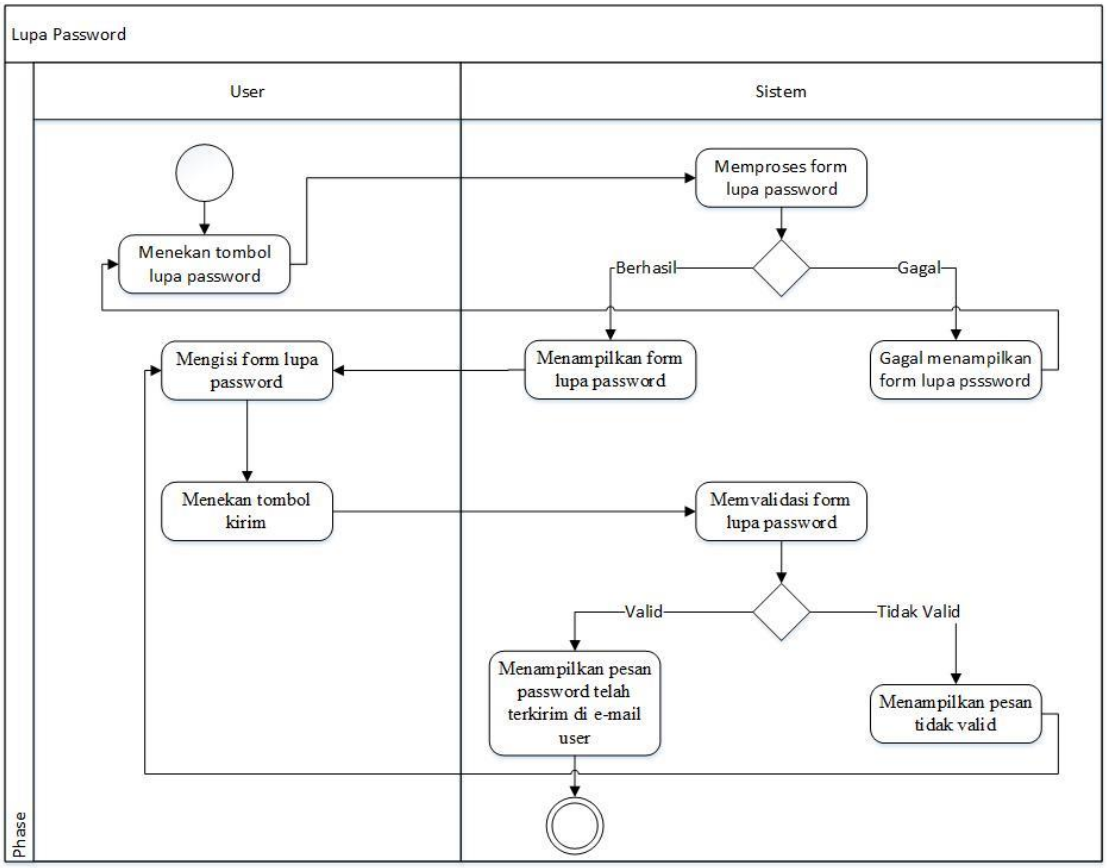
Berikut adalah Gambar 3.12 *Activity* Diagram *Login* yang menggambarkan *activity* diagram *login* :



Gambar 3.12 *Activity Diagram Login*

c) *Activity* Diagram Lupa *Password*

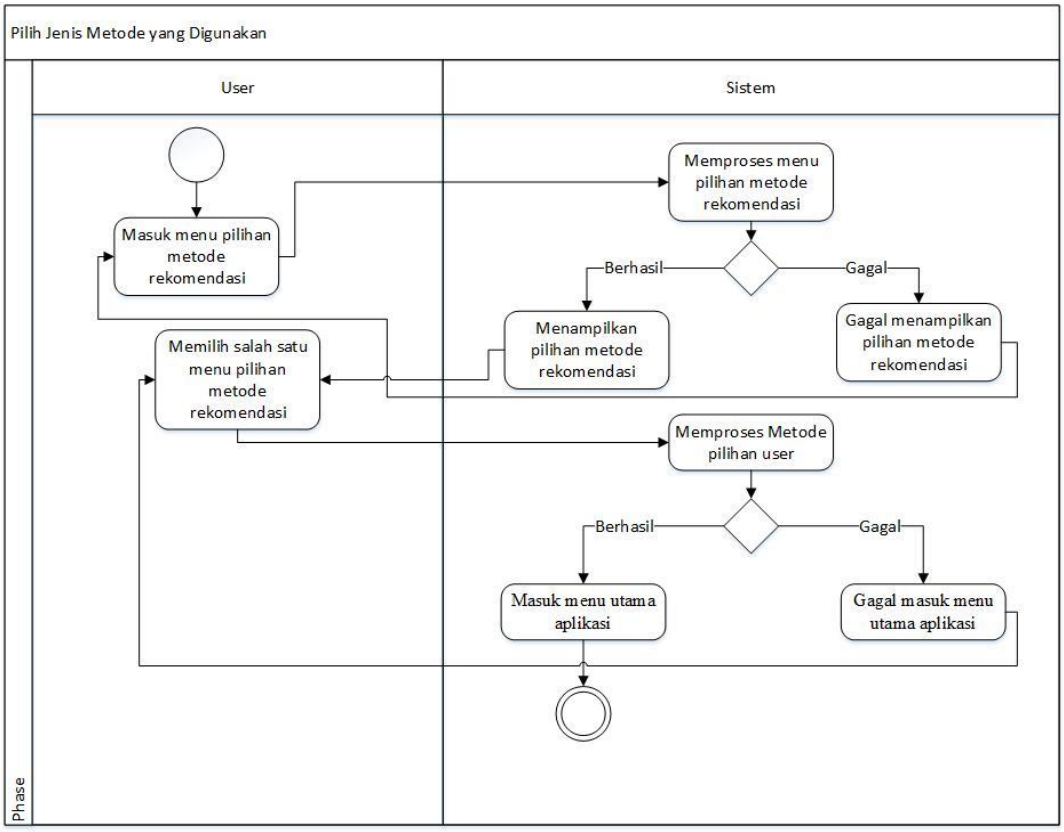
Berikut adalah Gambar 3.13 *Activity* Diagram Lupa *Password* yang menggambarkan *activity* diagram lupa *password* :



Gambar 3.13 *Activity Diagram* Lupa *Password*

d) *Activity* Diagram Pilih Jenis Metode yang Digunakan

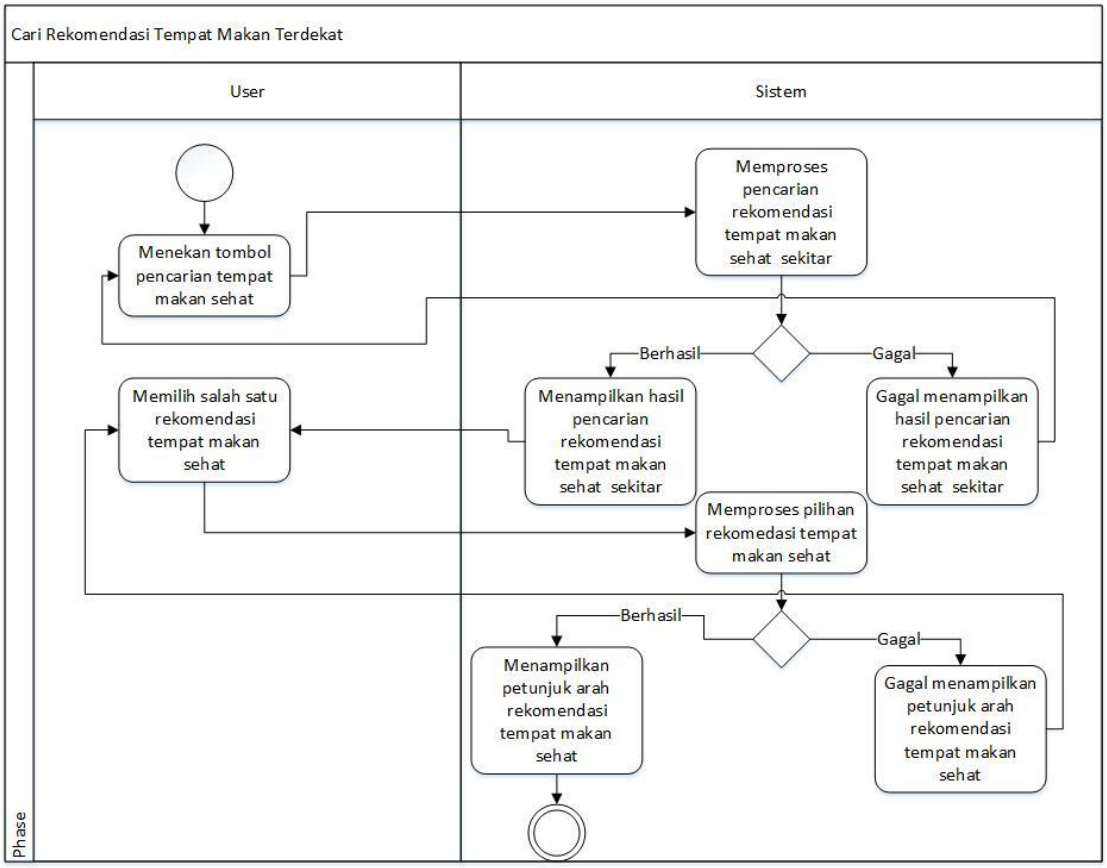
Berikut adalah Gambar 3.14 *Activity* Diagram Pilih Jenis Metode yang Digunakan yang menggambarkan *activity* diagram pilih jenis metode yang digunakan :



Gambar 3.14 *Activity Diagram* Pilih Jenis Metode yang Digunakan

e) *Activity* Diagram Cari Rekomendasi Tempat Makan Terdekat

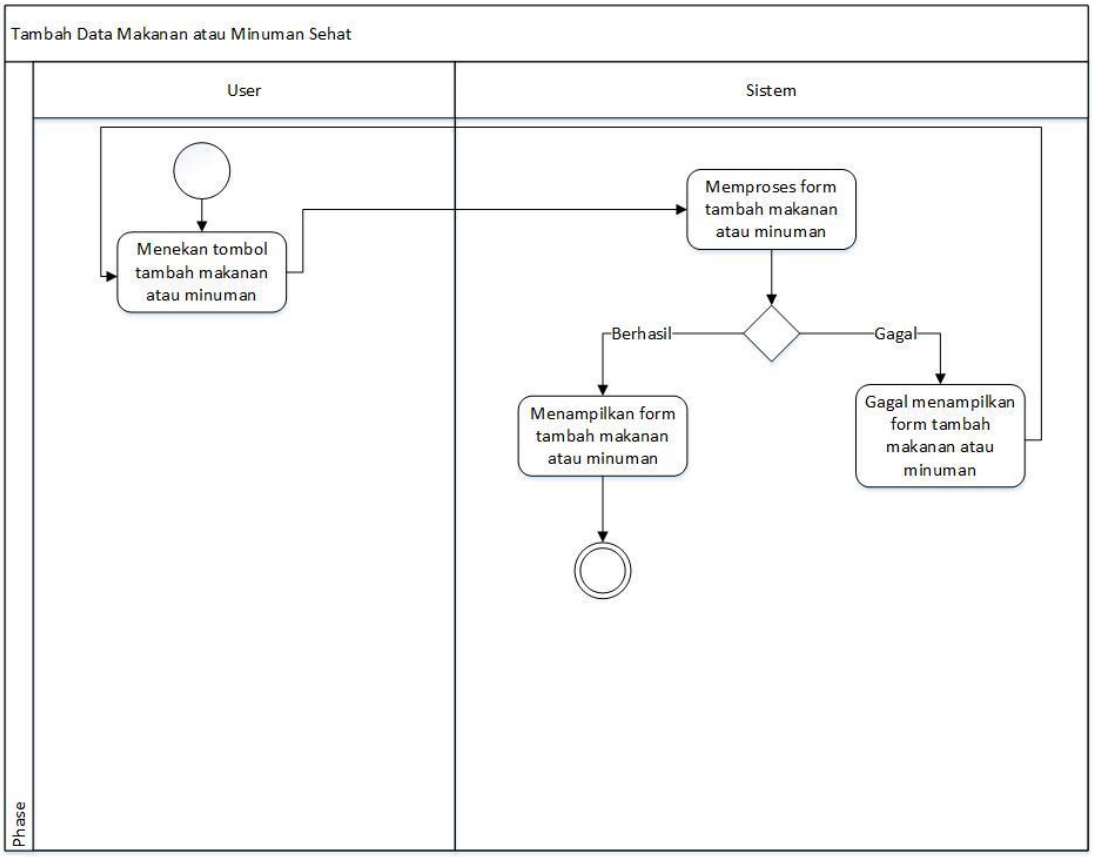
Berikut adalah Gambar 3.15 Activity Diagram Cari Rekomendasi Tempat Makan Terdekat yang menggambarkan *activity* diagram cari rekomendasi tempat makan terdekat :



Gambar 3.15 *Activity Diagram* Cari Rekomendasi Tempat Makan Terdekat

f) *Activity* Diagram Tambah Data Makanan atau Minuman Sehat

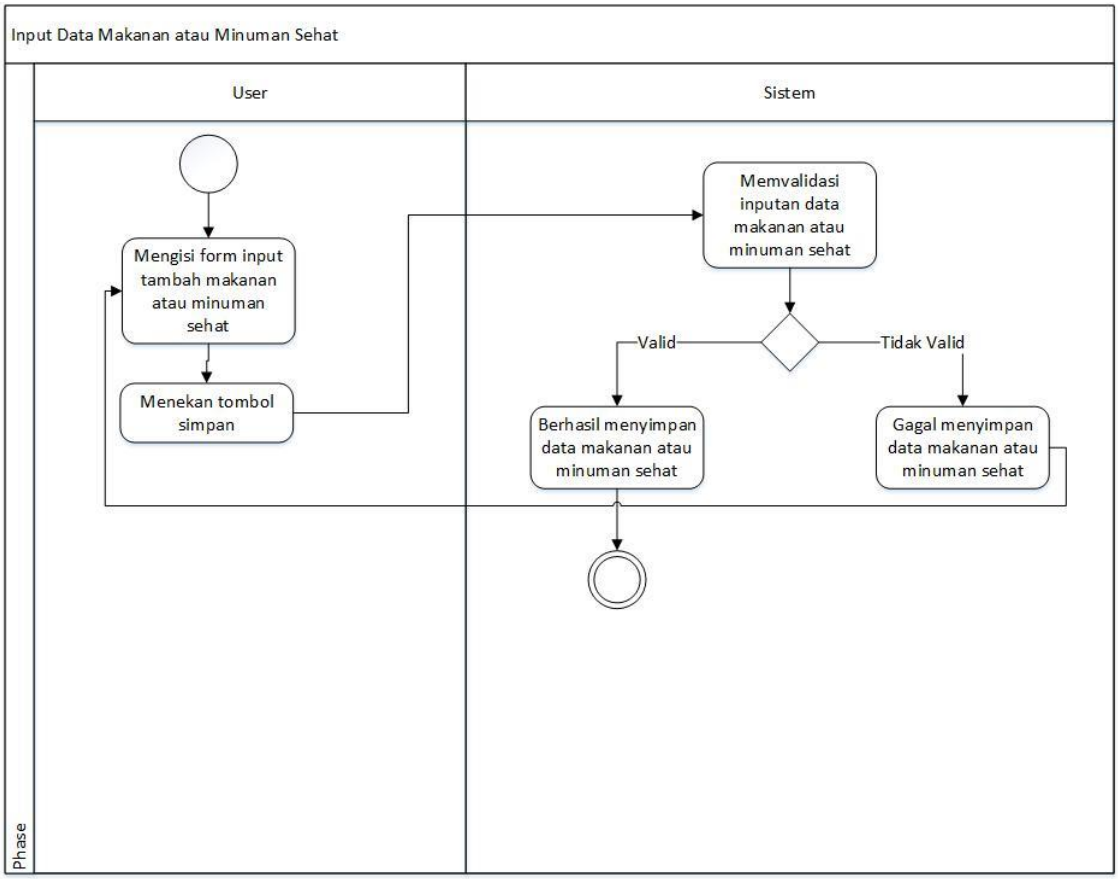
Berikut adalah Gambar 3.16 Activity Diagram Tambah Data Makanan atau Minuman Sehat yang menggambarkan *activity* diagram diagram tambah data makanan atau minuman sehat :



Gambar 3.16 *Activity Diagram* Tambah Data Makanan atau Minuman Sehat

g) *Activity* Diagram *Input* Data Makanan atau Minuman Sehat

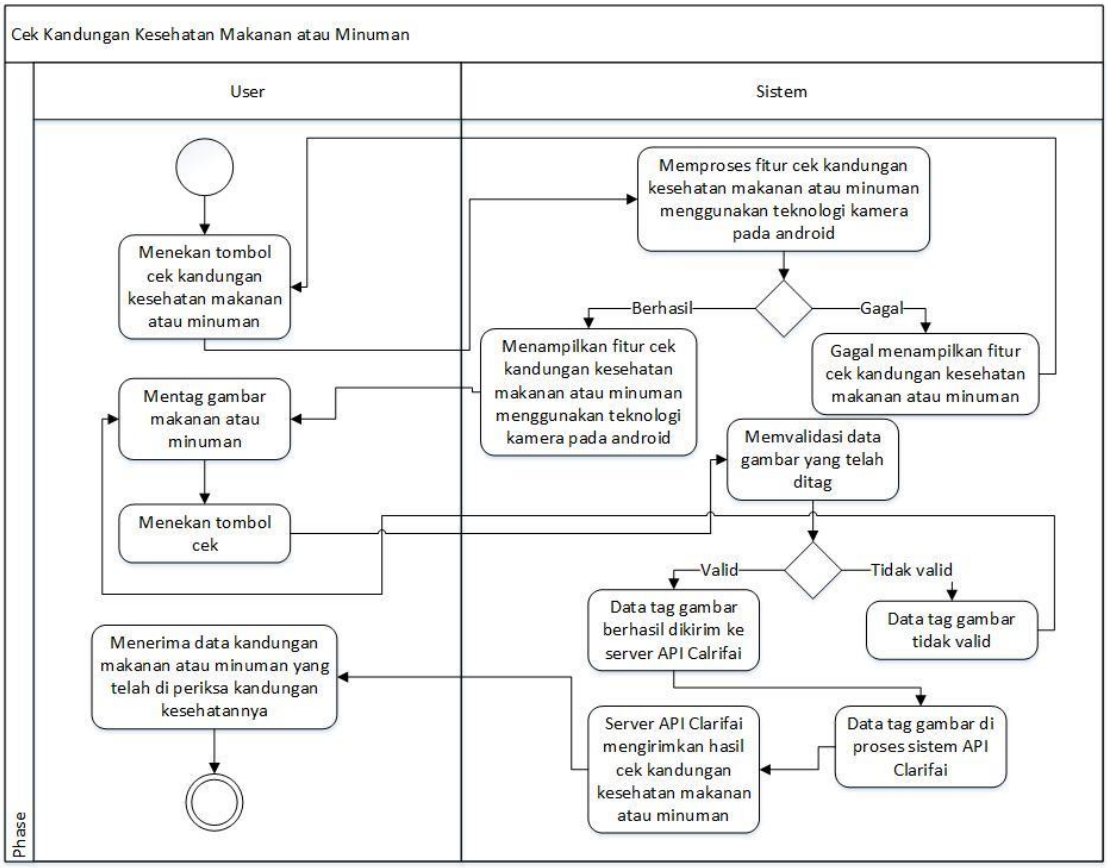
Berikut adalah Gambar 3.17 Activity Diagram *Input* Data Makanan atau Minuman Sehat yang menggambarkan *activity* diagram *input* data makanan atau minuman sehat :



Gambar 3.17 *Activity Diagram* Input Data Makanan atau Minuman Sehat

h) *Activity* Diagram Cek Kandungan Kesehatan Makanan atau Minuman

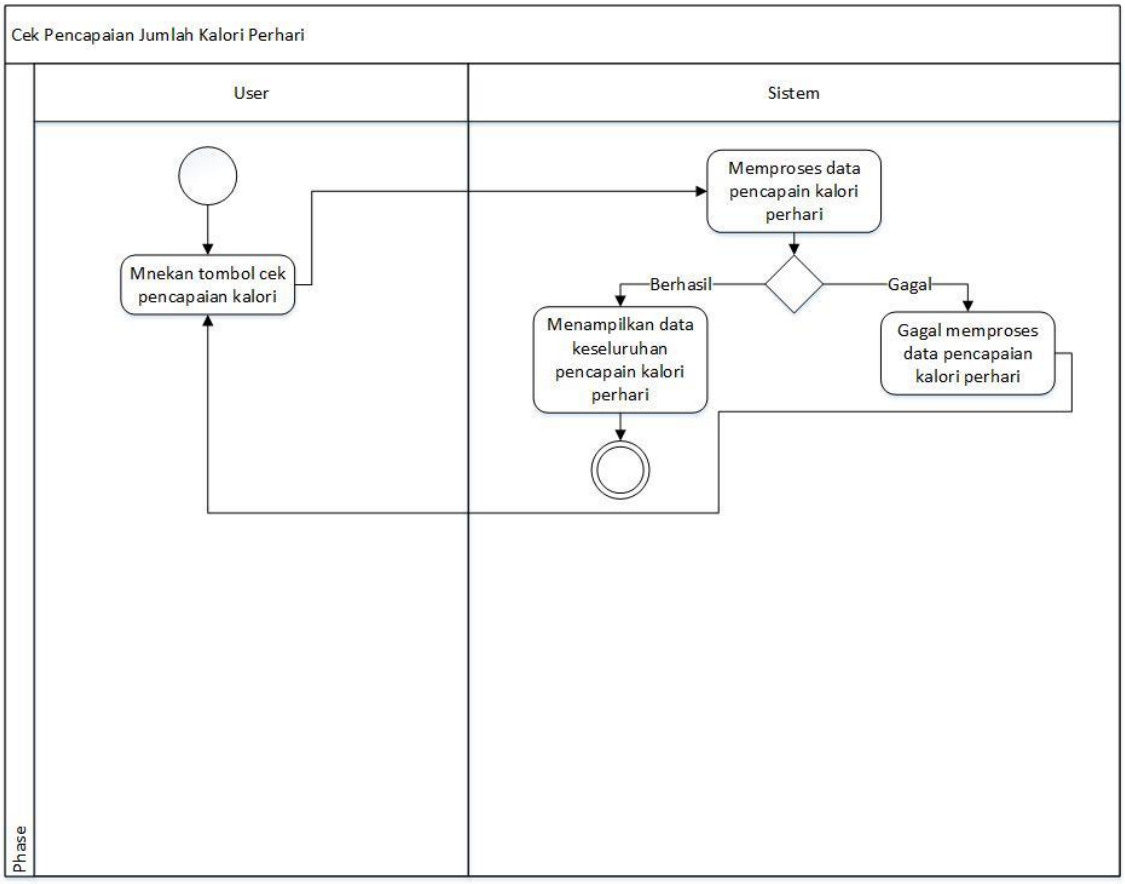
Berikut adalah Gambar 3.18 Activity Diagram Cek Kandungan Kesehatan Makanan atau Minuman yang menggambarkan *activity* diagram cek kandungan kesehatan makanan atau minuman :



Gambar 3.18 *Activity Diagram* Cek Kandungan Kesehatan Makanan atau Minuman

i) *Activity* Diagram Cek Pencapaian Jumlah Kalori Perhari

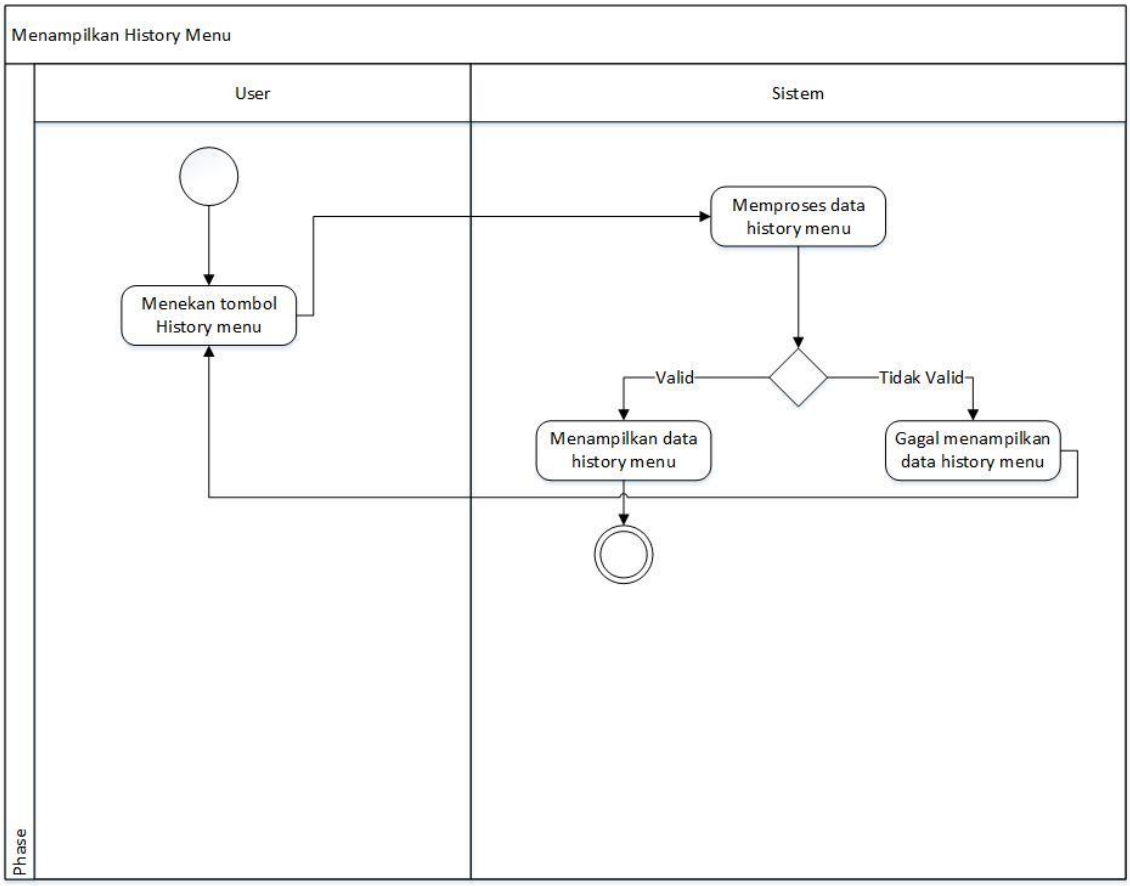
Berikut adalah Gambar 3.19 Activity Diagram Cek Pencapaian Jumlah Kalori Perhari yang menggambarkan *activity* diagram cek pencapaian jumlah kalori perhari :



Gambar 3.19 *Activity Diagram* Cek Pencapaian Jumlah Kalori Perhari

j) *Activity* Diagram Menampilkan History Menu

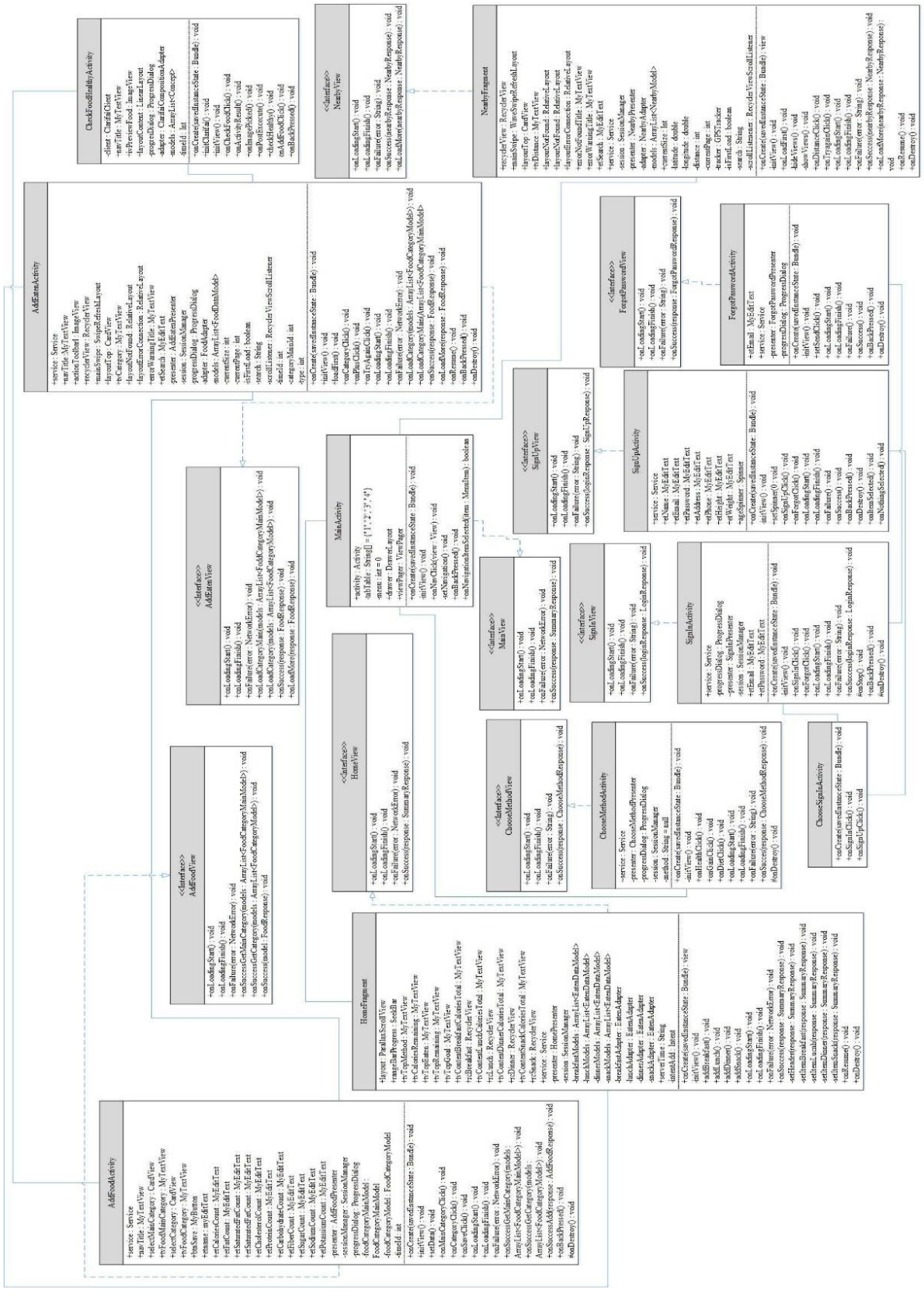
Berikut adalah Gambar 3.20 Activity Diagram Menampilkan History Menu yang menggambarkan *activity* diagram menampilkan history menu :



Gambar 3.20 *Activity Diagram* Menampilkan History Menu

3.1.7.5. *Class Diagram*

Class diagram digunakan untuk memberi gambaran suatu aplikasi. Dapat dilihat dari kelas-kelas yang ada dan cara kerja relasinya. Class diagram yang dibuat dapat dilihat pada gambar sebagai berikut :



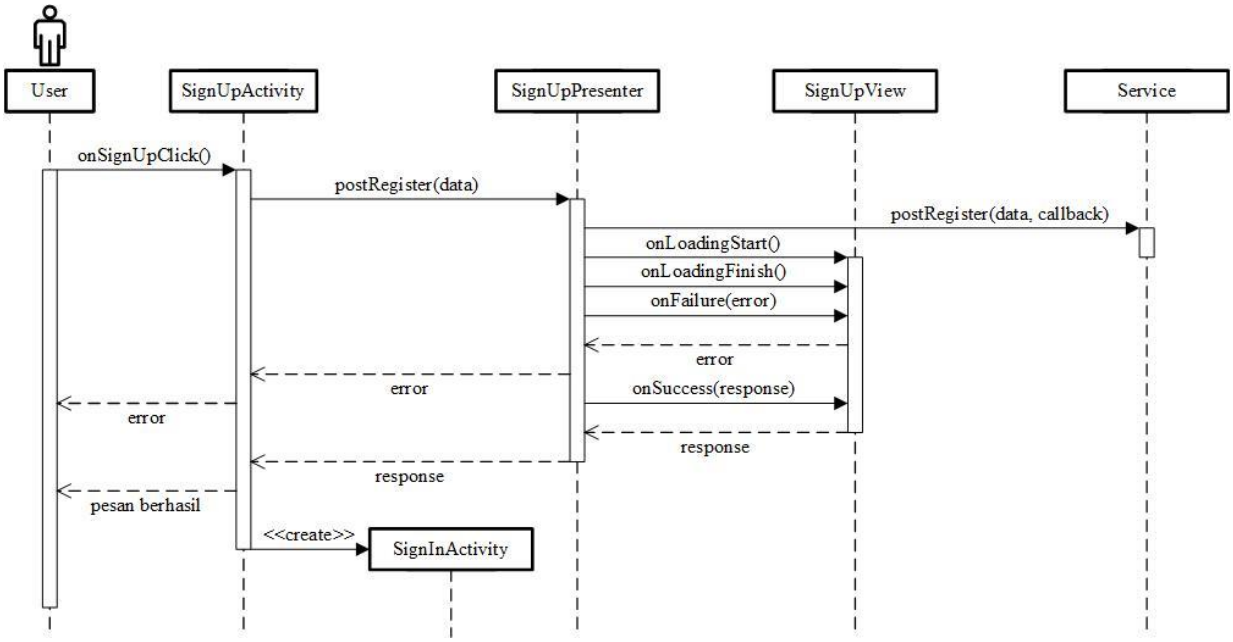
Gambar 3.21 ***Class Diagram***

3.1.7.6. *Sequence Diagram*

Sequence Diagram merupakan gambaran interkasi masing-masing objek pada setiap *Use Case* dalam urutan waktu. Interaksi ini berupa pengiriman serangakain data antar objek-objek yang saling berinteraksi. Berikut ini adalah sequence diagram yang dapat digambarkan, diantaranya adalah:

1. *Sequence Diagram* Daftar

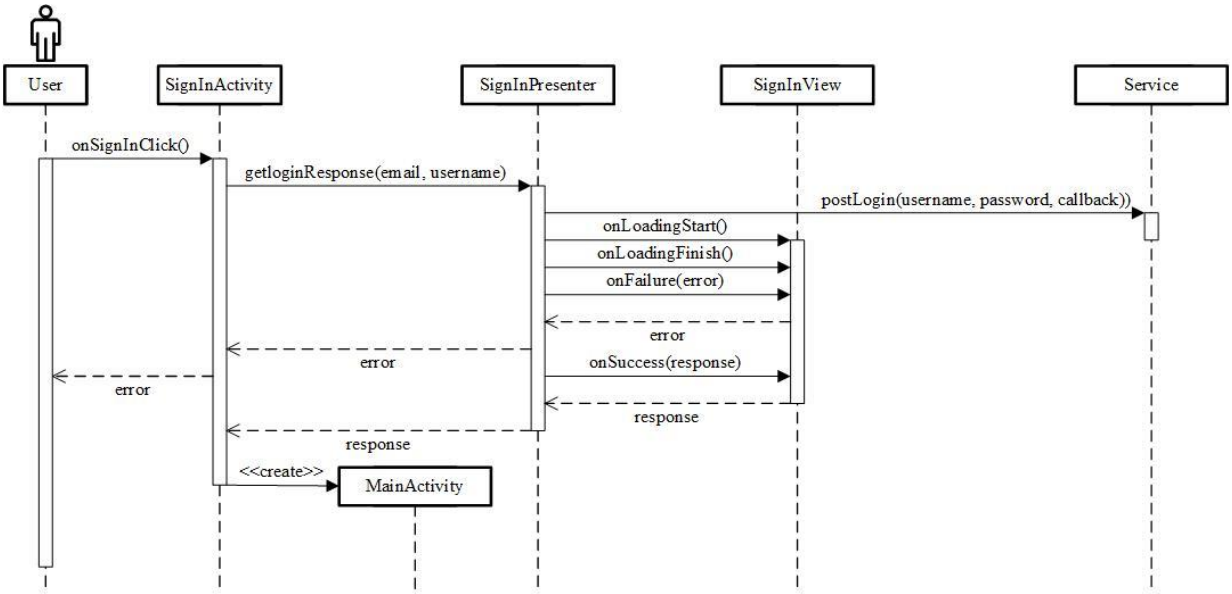
   Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses daftar. *Sequence Diagram* Daftar dapat dilihat pada Gambar 3.22 Sequence Diagram Daftar dibawah ini.



Gambar 3.22 *Sequence Diagram* Daftar

2. *Sequence Diagram Login*

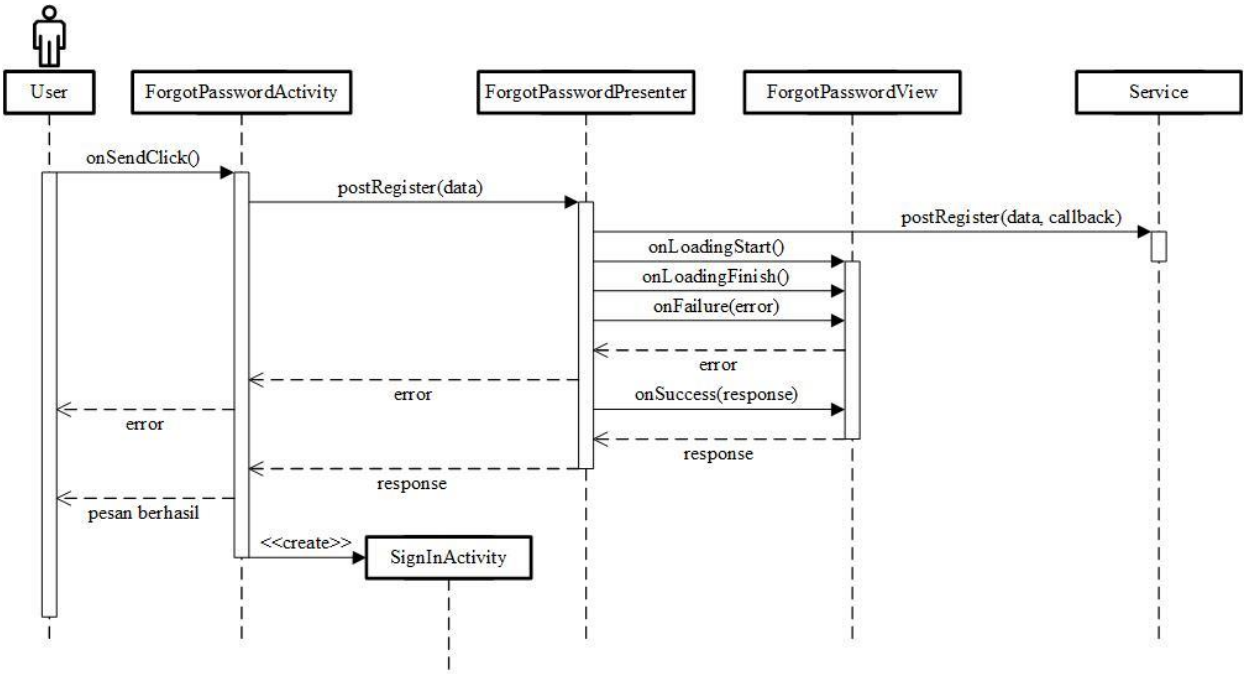
   Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses daftar. *Sequence Diagram Login* dapat dilihat pada Gambar 3.23 *Sequence Diagram Login* dibawah **ini.**



Gambar 3.23 *Sequence Diagram Login*

3. *Sequence Diagram* Lupa *Password*

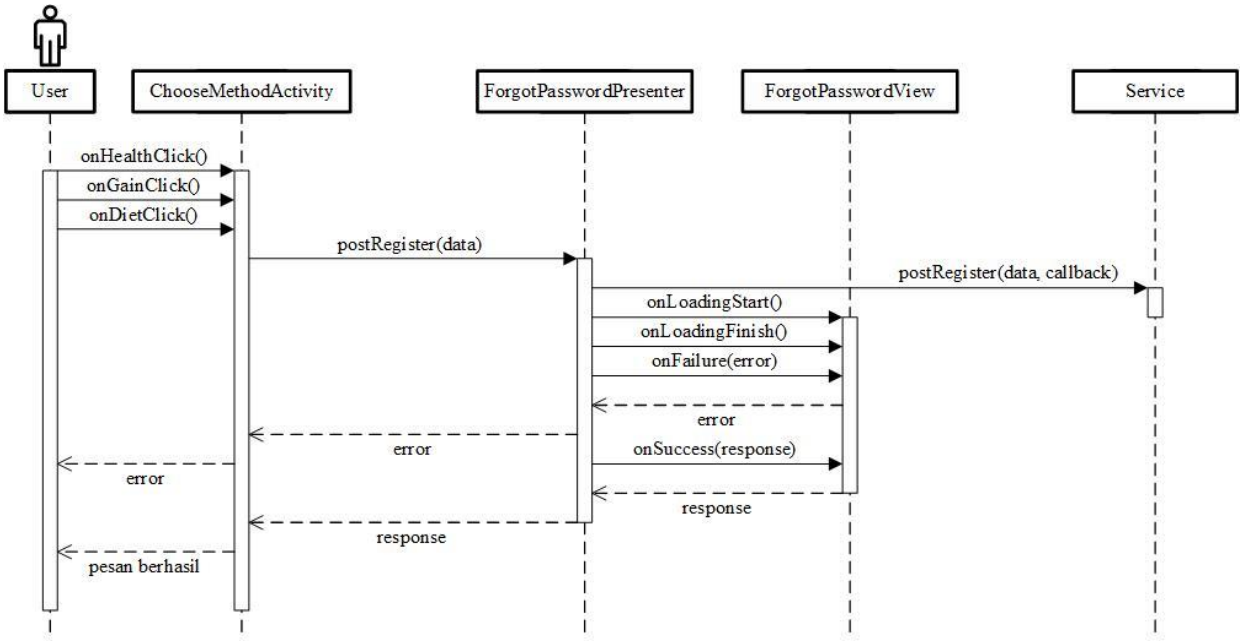
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses lupa *password*. *Sequence Diagram* lupa *password* dapat dilihat pada Gambar 3.24 *Sequence Diagram* Lupa *Password* dibawah ini.



Gambar 3.24 *Sequence Diagram* Lupa *Password*

4. *Sequence Diagram* Pilih Jenis Metode yang Digunakan

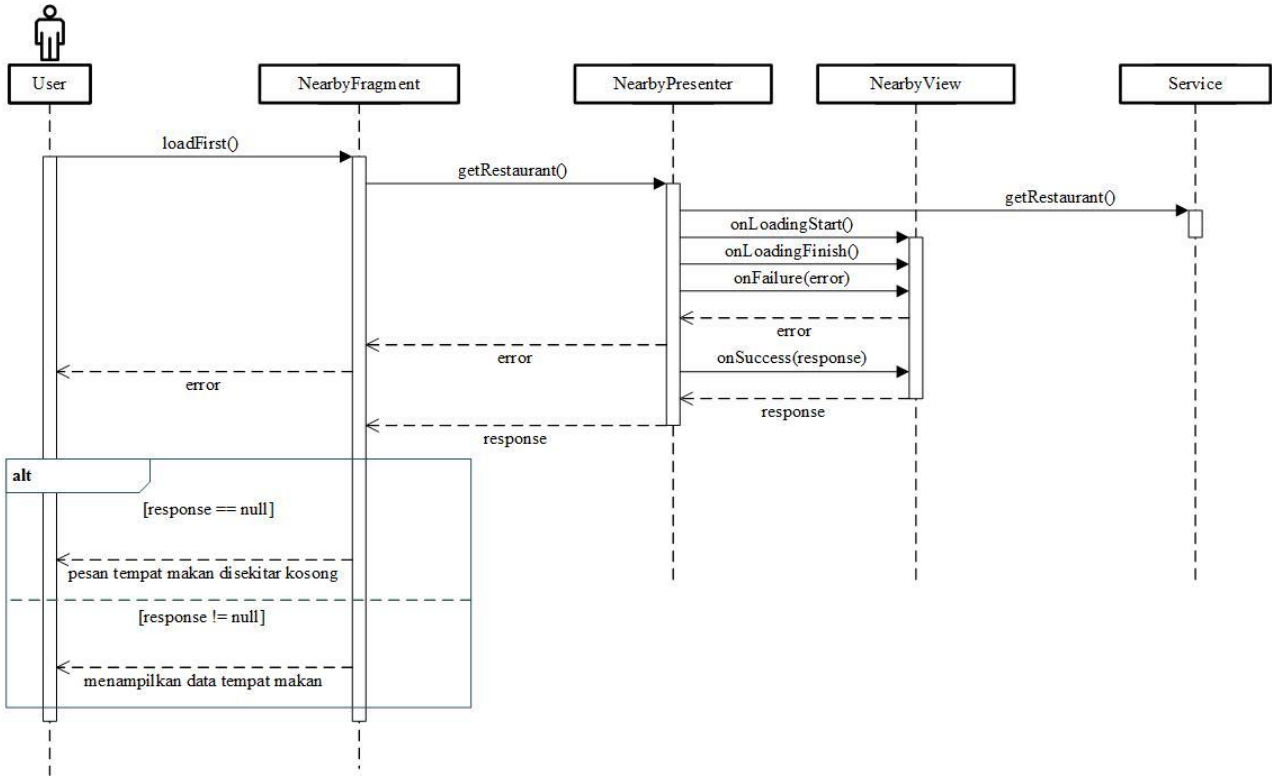
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses pilih jenis metode yang digunakan. *Sequence Diagram* Pilih Jenis Metode yang Digunakan dapat dilihat pada Gambar 3.25 *Sequence Diagram* Pilih Jenis Metode yang Digunakan dibawah ini.



Gambar 3.25 *Sequence Diagram* Pilih Jenis Metode yang Digunakan

5. *Sequence Diagram* Cari Rekomendasi Tempat Makan Terdekat

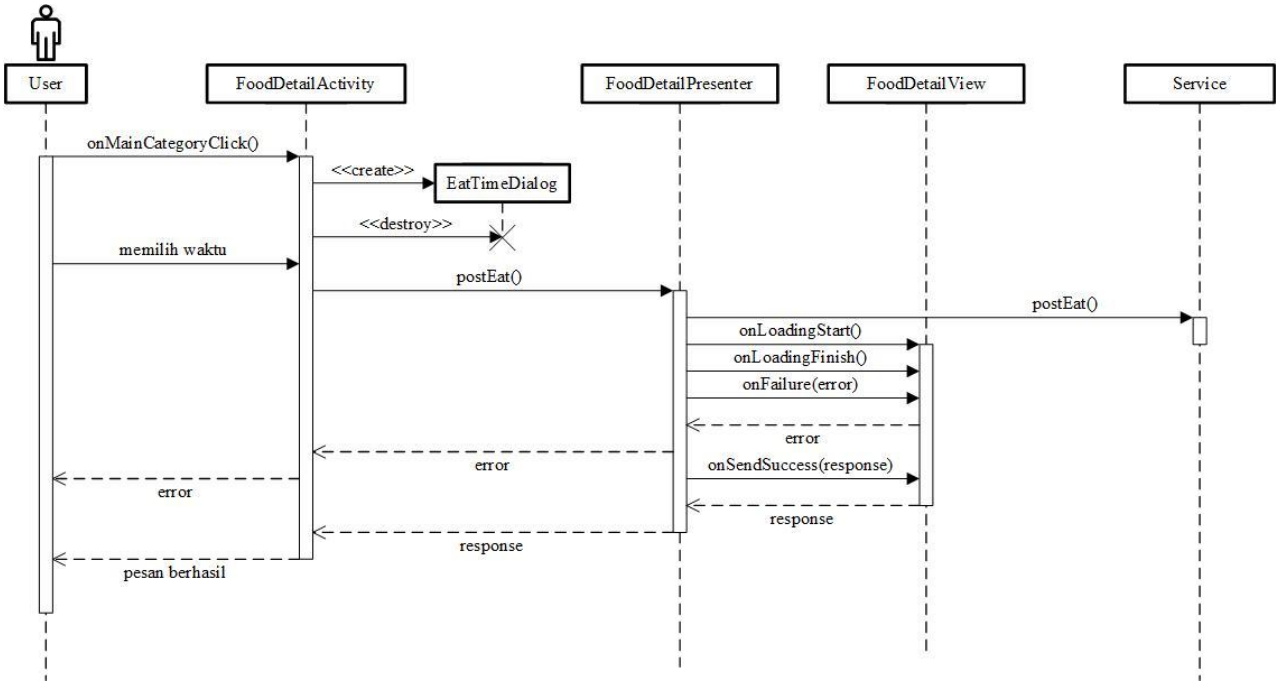
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses cari rekomendasi tempat makan terdekat. *Sequence Diagram* Cari Rekomendasi Tempat Makan Terdekat dapat dilihat pada Gambar 3.26 *Sequence Diagram* Cari Rekomendasi Tempat Makan Terdekat diabawah ini.



Gambar 3.26 *Sequence Diagram* Cari Rekomendasi Tempat Makan Terdekat

6. *Sequence Diagram* Tambah Data Makanan atau Minuman Sehat

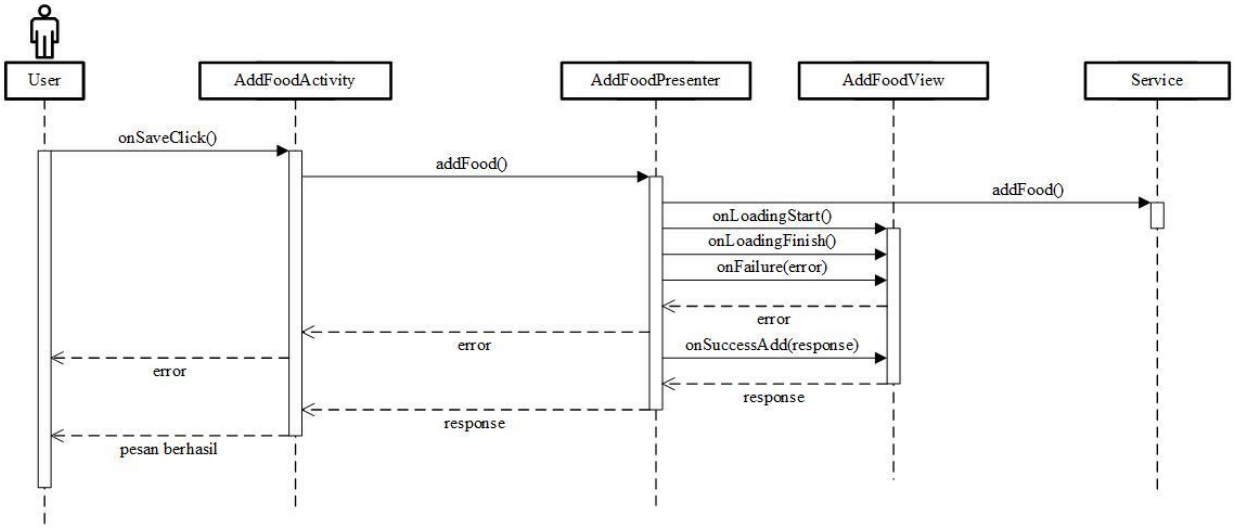
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses Tambah Data Makanan atau Minuman. *Sequence Diagram* Tambah Data Makanan atau Minuman Sehat dapat dilihat pada Gambar 3.27 *Sequence Diagram* Tambah Data Makanan atau Minuman Sehat dibawah ini.



Gambar 3.27 *Sequence Diagram* Tambah Data Makanan atau Minuman Sehat

7. *Sequence Diagram Input* Makanan atau Minuman Sehat

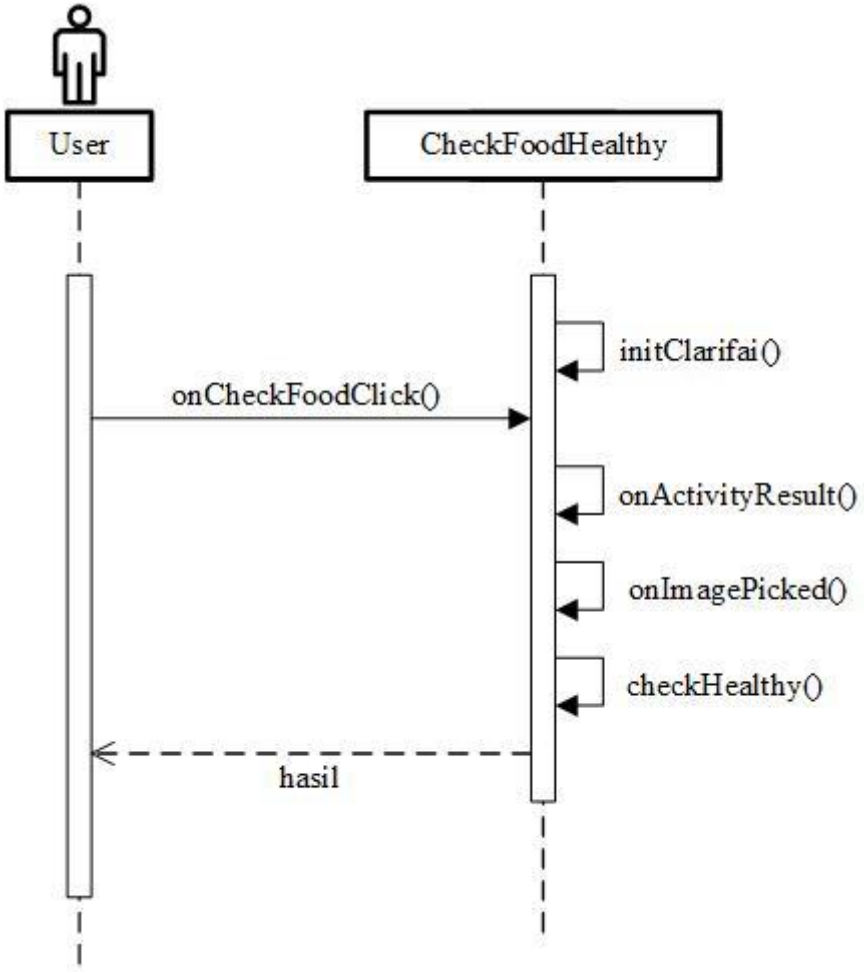
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses input data makanan atau minuman. *Sequence Diagram Input* Data Makanan atau Minuman Sehat dapat dilihat pada Gambar 3.28 *Sequence Diagram Input* Data Makanan atau Minuman Sehat dibawah ini.



Gambar 3.28 *Sequence Diagram Input* Data Makanan atau Minuman Sehat

8. *Sequence Diagram* Cek Kandungan Kesehatan Makanan atau Minuman

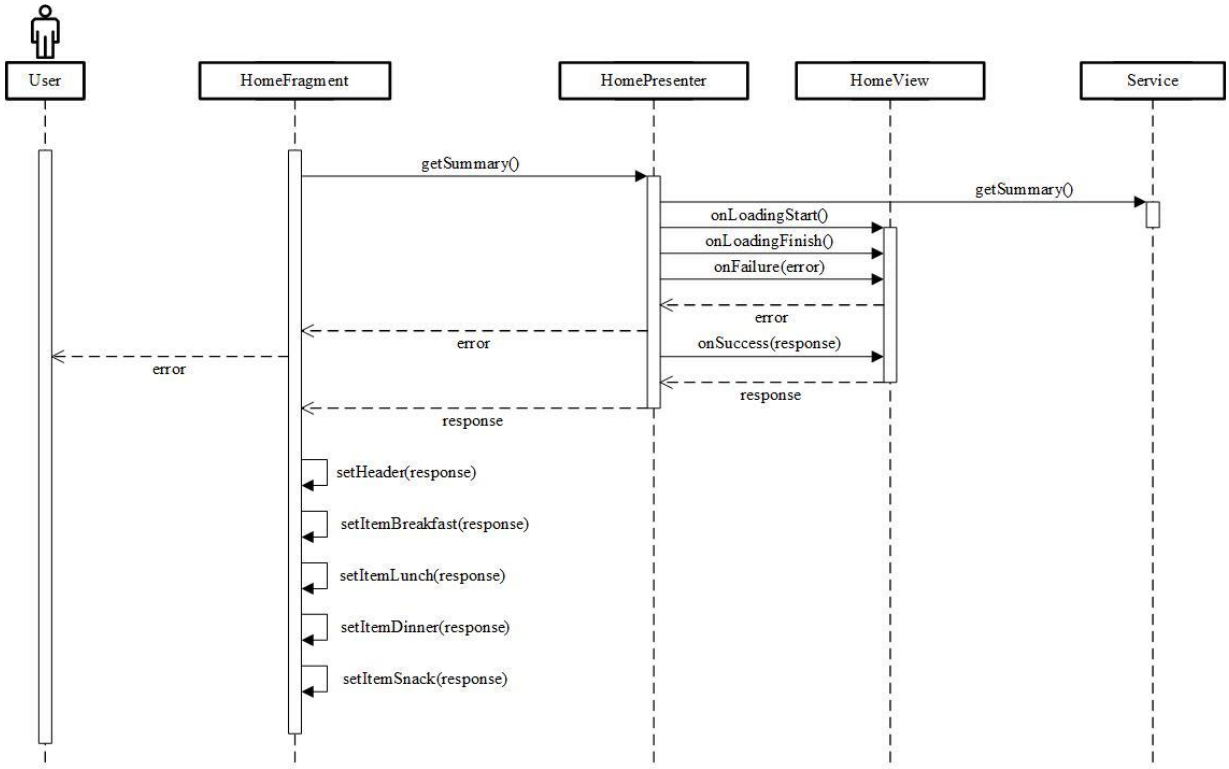
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses cek kandungan kesehatan makanan atau minuman. *Sequence diagram* Cek Kandungan Kesehatan Makanan atau Minuman dapat dilihat pada Gambar 3.29 *Sequence Diagram* Cek Kandungan Kesehatan Makanan atau Minuman dibawah ini.



Gambar 3.29 *Sequence Diagram* Cek Kandungan Kesehatan Makanan atau Minuman

9. *Sequence Diagram* Cek Pencapaian Jumlah Kalori Perhari

Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses cek pencapaian kalori perhari. *Sequence Diagram* cek pencapaian kalori perhari dapat dilihat pada Gambar 3.30 *Sequence Diagram* Cek Pencapaian Kalori Perhari.



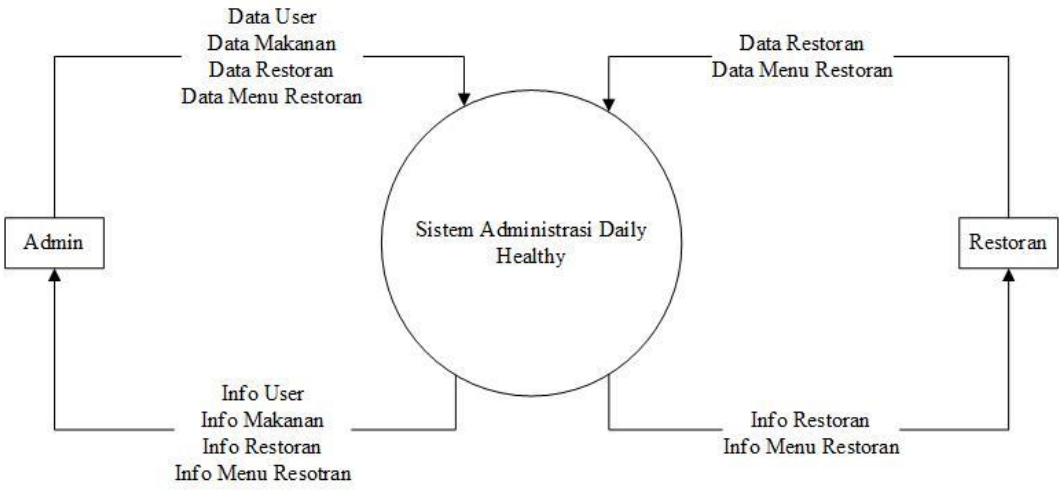
Gambar 3.30 *Sequence Diagram* Cek Pencapaian Kalori Perhari

3.1.8. Analisis Kebutuhan Fungsional *WEB*

Analissi kebutuhan fungsional di *platform web* menggunakan analisis terstruktur yang meliputi *Diagram* Konteks, *Data Flow Diagram*, Spesifikasi Proses dan Kamus data.

3.1.8.1. *Diagram* Konteks

Diagram konteks merupakan salah satu alat banu dalam melaukan analisis terstruktur. Diagramm kontek ini menggambarkan suatu sistem secara garis besarnya dan keseluruhannya saja. Dalam diagram konteks juga digambarkan entitas eksternal yang merupakan *brainware* yang menghasilkan data yang akan diolah sistem maupun tujuan dari informasi yang dihasilkan oleh sistem.



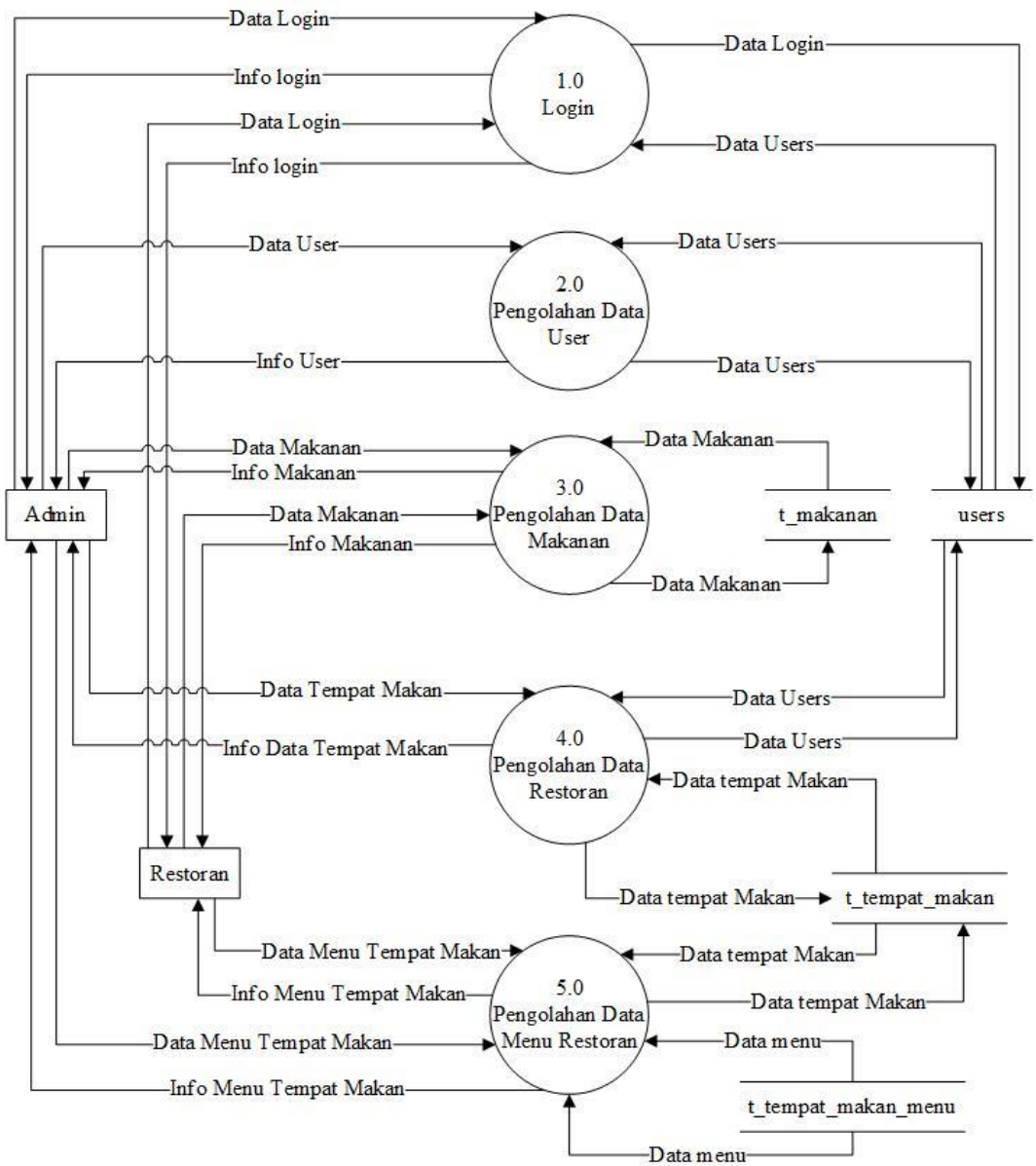
Gambar 3.31 *Diagram* Konteks

3.1.8.2. *Data Flow Diagram*

Data Flow Diagram merupaakn sebuah representasi dari suatu sistem yang menggambarkan bagian-bagian dari sistem yang akan dibangun beserta keterkaitan antara bagian-bagian yang ada. Dari *Data Flow Diagram* ini seseorang bisa mengetahui sumber dari informasi di dalam sistem maupun tujuan dari masukan yang berasal dari entitas eksternal. Adapun *Data Flow Diagram* pada pembangunan sistem di *platform web*site adalah sebagai berikut.

1. *Data Flow Diagram* Level 1

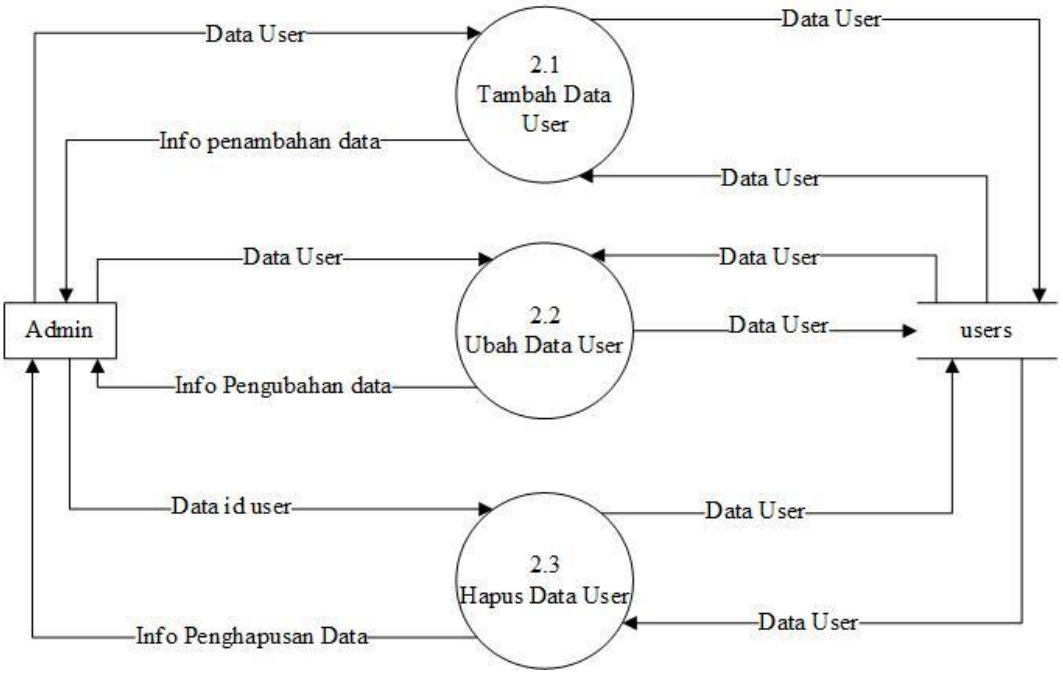
   Berikut ini adalah *data flow diagram* level 1 ditunjukan pada Gambar 3.32 DFD Level 1 dibawah ini.



Gambar 3.32 DFD Level 1

2. *Data Flow Diagram* Level 2 Pengolahan Data *User*

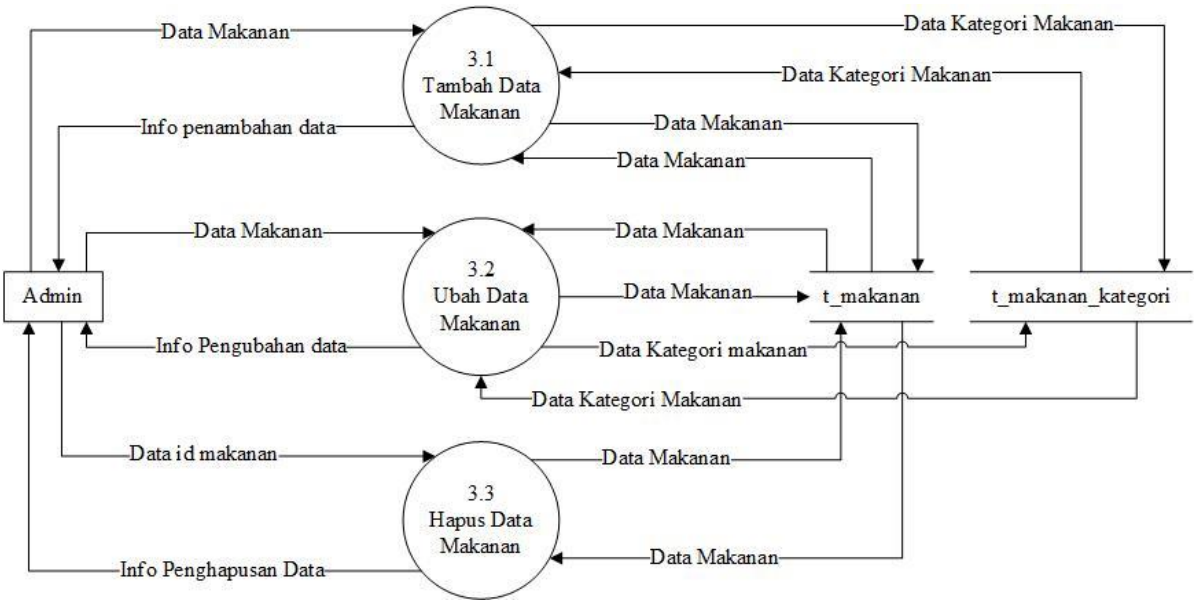
Berikut ini adalah *data flow diagram* level 2 pengolahan data makanan ditunjukan pada Gambar 3.33 DFD Level 2 Proses Pengolahan Data *User* dibawah ini.



Gambar 3.33 DFD Level 2 Pengolahan Data User

3. *Data Flow Diagram* Level 2 Pengolahan Data Makanan

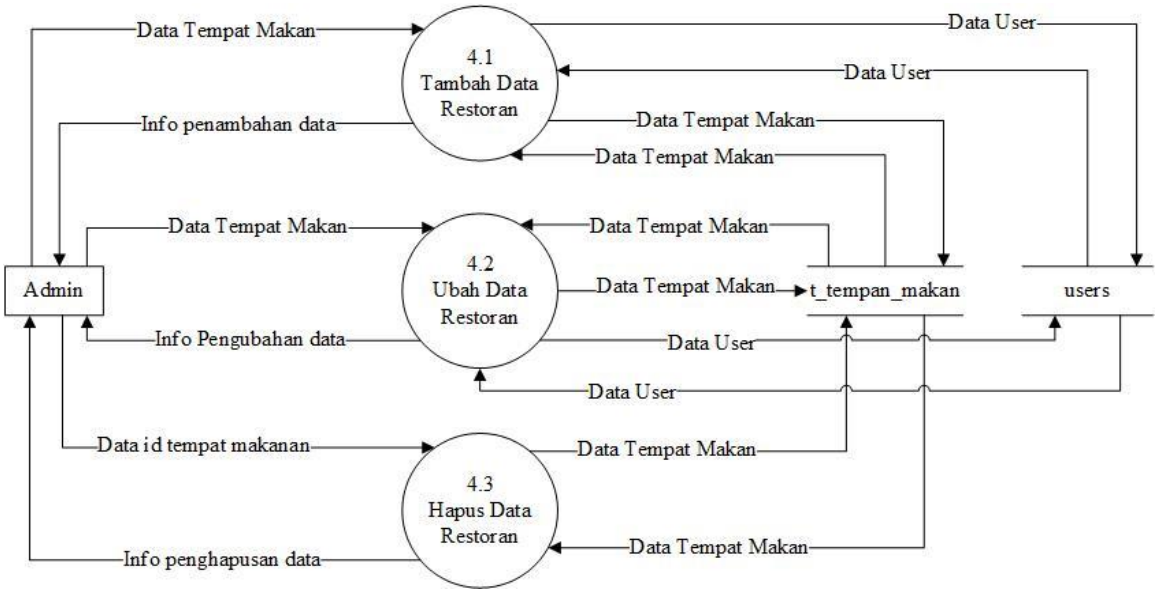
Berikut ini adalah *data flow diagram* level 2 tambah data restroan ditunjukan pada Gambar 3.34 DFD Level 2 Proses Tambah Data Makanan dibawah ini.



Gambar 3.34 DFD Level 2 Proses Tambah Data Makanan

4. *Data Flow Diagram* Level 2 Tambah Data Restoran

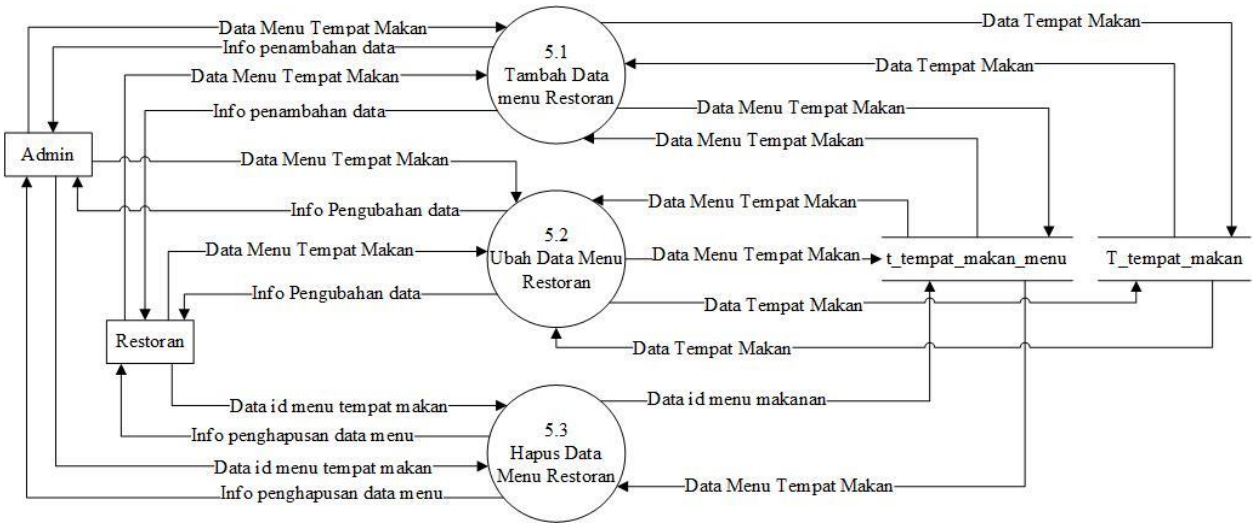
Berikut ini adalah *data flow diagram* level 2 proses tambah data menu restoran ditunjukan pada Gambar 3.35 DFD Level 2 Proses Tambah Data Restoran dibawah ini.



Gambar 3.35 DFD Level 2 Proses Tambah Data Restoran

5. *Data Flow Diagram* Level 2 Pengolahan Data Menu Restoran

Berikut ini adalah *data flow diagram* level 2 proses tambah data menu restoran ditunjukan pada Gambar 3.36 DFD Level 2 Pengolahan Data Menu Restoran dibawah ini.



Gambar 3.36 DFD Level 2 Pengolahan Data Menu Restoran

3.1.8.3. Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan proses model aliran yang terdapat pada *Data Flow Duagram* (DFD). Berikut ini spesifikasi proses pada pembangunan sistem di *platform web*site.

1. Spesifikasi Proses Level 1

   Spesifikasi proses level 1 digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD level 1.

Tabel 3.21 Sepsifikasi Proses Level 1

No	Proses	Keterangan
	No. Proses	1
	Nama Proses	*Login*
	Source	*Admin*
	Input	Data *Login*
	Output	Info *Login*
	Destination	*Admin*
	Logika Proses	1. *Admin* memasukaan data *username* dan *password* pada halaman *login*. 2. Sistem melalukan validasi *username* dan *password user*. 3. Apabila *login* valid, sistem akan menampilkan halaman utama. 4. Apabila *login* tidak valid, sistem akan menampilkan pesan kesalahan.
	No. Proses	2
	Nama Proses	Pengolahan Data *User*
	Source	*Admin*
	Input	Data *User*
	Output	Info *User*
	Destination	*Admin*
	Logika Proses	1. *Admin* Memilih menu data *user* 2. Dalam halaman data *user* terdapat menu tambah data *user*, ubah data *user* dan hapus data *user*
	No. Proses	3
	Nama Proses	Pengolahan Data Makanan
	Source	*Admin*
	Input	Data Makanan
	Output	Info Maknan
	Destination	*Admin*
	Logika Proses	1. *Admin* memilih menu data makanan 2. Dalam halaman data makanan terdapat menu tambah data makanan, ubah data makanan dan hapus data makanan.
	No. Proses	4
	Nama Proses	Pengolahan Data Restoran
	Source	*Admin*
	Input	Data Tempat Makan
	Output	Info Tempat Makan
	Destination	*Admin*
	Logika Proses	1. *Admin* memilih menu data restoran 2. Dalam halaman data restoran terdapat menu tambah data

		restoran, ubah data restoran dan hapus data resotoran
	No. Proses	5
	Nama Proses	Pengolahan Data Menu Restoran
	Source	*Admin*, Restoran
	Input	Data Menu Tempat Makan
	Output	Info Menu Tempat Makan
	Destination	*Admin*, Restoran
	Logika Proses	1. *Admin* atau restoran memilih menu data menu restoran 2. Dalam menu data menu restoran terdapat menu tambah data menu restoran, ubah data menu restoran dan hapus data menu restoran

2. Spesifikasi Proses Level 2 Pengolahan Data *User*

Spesifikasi proses level 2 pengolahan data *user* digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD level 2 proses Pengolahan Data *User*.

Tabel 3.22 Spesifikasi Proses Level 2 Pengolahan Data User

No	Proses	Keterangan
	No. Proses	2.1
	Nama Proses	Tambah Data *User*
	Source	*Admin*
	Input	Data *User*
	Output	Info Penambahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* memasukkan data *user* 2. Sistem menyimpan data *user* kedalam *database* 3. Apabila sistem berhasil menyimpan data *user* maka sistem akan menampilkan pesan berhasil 4. Apabila sistem gagal menyimpan data *user* maka sistem akan menampilkan pesan gagal
	No. Proses	2.2
	Nama Proses	Ubah data *User*
	Source	*Admin*
	Input	Data *User*
	Output	Info Pengubahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* mengubah data *user*. 2. Sistem mengubah data *user* didalam *database* sesuai dengan data *user* yang diubah. 3. Apabila sistem berhasil mengubah data *user*, maka sistem akan menampilkan data *user* yang telah di ubah dengan disertai pesan berhasil. 4. Apabila sistem gagal mengubah data *user*, maka sistem akan menampilkan halaman data *user* dengan disertai pesan kesalahan.
	No. Proses	2.3
	Nama Proses	Hapus Data *User*
	Source	*Admin*
	Input	Data id *User*
	Output	Info penghapusan data
	Destination	*Admin*

	Logika Proses	1. *Admin* memilih data *user* yang akan dihapus. 2. Sistem menampilkan pesan konfirmasi penghapusan 3. Sistem menghapus data *user* dari *database* 4. Apabila data berhasil dihapus maka sistem akan menampilkan pesan berhasil 5. Apabila data gagal dihapus maka sistem akan menampilkan pesan kesalahan.

3. Spesifikasi Proses Level 2 Proses Pengolahan Data Makanan

Spesifikasi proses level 2 proses pengolahan data makanan digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD level 2 Proses Pengolahan Data Makanan.

Tabel 3.23 Spesifikasi Proses Level 2 Pengolahan Data Makanan

No	Proses	Keterangan
	No. Proses	3.1
	Nama Proses	Tambah Data Makanan
	Source	*Admin*
	Input	Data id makanan
	Output	Info Penambahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* memasukkan data makanan 2. Sistem menyimpan data makanan kedalam *database* 3. Apabila sistem berhasil menyimpan data makanan maka sistem akan menampilkan pesan berhasil 4. Apabila sistem gagal menyimpan data makanan maka sistem akan menampilkan pesan gagal
	No. Proses	3.2
	Nama Proses	Ubah data Makanan
	Source	*Admin*
	Input	Data Makanan
	Output	Info Pengubahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* mengubah data makanan. 2. Sistem mengubah data makanan didalam *database* sesuai dengan data makanan yang diubah. 3. Apabila sistem berhasil mengubah data makanan, maka sistem akan menampilkan data makanan yang telah di ubah dengan disertai pesan berhasil. 4. Apabila sistem gagal mengubah data makanan, maka sistem akan menampilkan halaman data makanan dengan disertai pesan kesalahan.
	No. Proses	3.3
	Nama Proses	Hapus Data Makanan
	Source	*Admin*
	Input	Data Makanan
	Output	Info penghapusan data
	Destination	*Admin*
	Logika Proses	1. *Admin* memilih data makanan yang akan dihapus. 2. Sistem menampilkan pesan konfirmasi penghapusan 3. Sistem menghapus data makanan dari *database*

		4. Apabila data berhasil dihapus maka sistem akan menampilkan pesan berhasil 5. Apabila data gagal dihapus maka sistem akan menampilkan pesan kesalahan.

4. Spesifikasi Proses Level 2 Proses Pengolahan Data Restoran

Spesifikasi proses level 2 proses pengolahan data restoran digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD level 2 Proses Pengolahan Data Restoran.

Tabel 3.24 Spesifikasi Proses Level 2 Pengolahan Data Restoran

No	Proses	Keterangan
	No. Proses	4.1
	Nama Proses	Tambah Data Restoran
	Source	*Admin*
	Input	Data Tempat Makan
	Output	Info Penambahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* memasukkan data tempat makan 2. Sistem menyimpan data tempat makan kedalam *database* 3. Apabila sistem berhasil menyimpan data tempat makan maka sistem akan menampilkan pesan berhasil 4. Apabila sistem gagal menyimpan data tempat makan maka sistem akan menampilkan pesan gagal
	No. Proses	4.2
	Nama Proses	Ubah data Resotran
	Source	*Admin*
	Input	Data Tempat Makan
	Output	Info Pengubahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* mengubah data Tempat Makan. 2. Sistem mengubah data tempat makan didalam *database* sesuai dengan data restoran yang diubah. 3. Apabila sistem berhasil mengubah data tempat makan, maka sistem akan menampilkan data tempat makan yang telah di ubah dengan disertai pesan berhasil. 4. Apabila sistem gagal mengubah data tempat makan, maka sistem akan menampilkan halaman data tempat makan dengan disertai pesan kesalahan.
	No. Proses	4.3
	Nama Proses	Hapus Data Restoran
	Source	*Admin*
	Input	Data id Tempat Makan
	Output	Info penghapusan data
	Destination	*Admin*
	Logika Proses	1. *Admin* memilih data tempat makan yang akan dihapus. 2. Sistem menampilkan pesan konfirmasi penghapusan 3. Sistem menghapus data tempat makan dari *database* 4. Apabila data berhasil dihapus maka sistem akan menampilkan pesan berhasil

		5. Apabila data gagal dihapus maka sistem akan menampilkan pesan kesalahan.

5. Spesifikasi Proses Level 2 Proses Pengolahan Data Menu Restoran

Spesifikasi proses level 2 proses pengolahan data menu restoran digunakan untuk menggambarkan proses model aliran yang terdapat pada DFD level 2 Proses Pengolahan Data Menu Restoran.

Tabel 3.25 Spesifikasi Proses Level 2 Pengolahan Data Menu Restoran

No	Proses	Keterangan
	No. Proses	5.1
	Nama Proses	Tambah Data Menu Restoran
	Source	*Admin*
	Input	Data Menu Resotoran
	Output	Info Penambahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* memasukkan data menu tempat makan 2. Sistem menyimpan data menu tempat makan kedalam *database* 3. Apabila sistem berhasil menyimpan data menu tempat makan maka sistem akan menampilkan pesan berhasil 4. Apabila sistem gagal menyimpan data menu tempat makana maka sistem akan menampilkan pesan gagal
	No. Proses	5.2
	Nama Proses	Ubah data Menu Resotran
	Source	*Admin*
	Input	Data Menu Tempat Makan
	Output	Info Pengubahan Data
	Destination	*Admin*
	Logika Proses	1. *Admin* mengubah data Menu Tempat Makan. 2. Sistem mengubah data menu tempat makan didalam *database* sesuai dengan data tempat makana yang diubah. 3. Apabila sistem berhasil mengubah data menu tempat makan, maka sistem akan menampilkan data menu tempat makan yang telah di ubah dengan disertai pesan berhasil. 4. Apabila sistem gagal mengubah data menu tempat makan, maka sistem akan menampilkan halaman data menu tempat makan dengan disertai pesan kesalahan.
	No. Proses	4.3
	Nama Proses	Hapus Data Menu Restoran
	Source	*Admin*
	Input	Data id Menu Tempat Makan
	Output	Info penghapusan data
	Destination	*Admin*
	Logika Proses	1. *Admin* memilih data menu tempat makan yang akan dihapus. 2. Sistem menampilkan pesan konfirmasi penghapusan 3. Sistem menghapus data menu tempat makan dari *database* 4. Apabila data berhasil dihapus maka sistem akan menampilkan pesan berhasil 5. Apabila data gagal dihapus maka sistem akan menampilkan pesan kesalahan.

3.1.8.4. Kamus Data

Kamus data dibuat berdasarkan arus data yang mengalir pada diagram konteks dan data flow diagram, sebgaimana diperlihatkan pada

Tabel 3.26 Kamus Data

Nama Aliran Data	**Data *Login***
Where used/how used	Proses 1
Deskripsi	Berisi data *login admin* dan restoran untuk masuk kedalam sistem
Struktur data	*email* + *password*
email	[A-Z|a-z|0-9|@|.|-]
password	[A-Z|a-z|0-9]
Nama Aliran Data	**Data *User***
Where used/how used	Proses 2, Proses 3.1, Proses 3.2, Proses 4.1, Proses 4.2
Deskripsi	Berisi Data *user*s *user* aplikasi
Struktur data	name + age + *email* + *password* + address + phone + height + weight + plan + image
name	[A-Z|a-z]
age	[0-9]
email	[A-Z|a-z|0-9]
password	[A-Z|a-z|0-9]
address	[A-Z|a-z|0-9]
phone	[0-9]
height	[0-9]
weight	[0-9]
plan	[A-Z|a-z]
image	[A-Z|a-z|0-9]
Nama Aliran Data	**Data Makanan**
Where used/how used	Proses 3.0, Proses 3.1, Proses 3.2
Deskripsi	Berisi data makanan
Struktur data	id, nama makanan
id	[0-9]
nama_makanan	[A-Z|a-z|0-9|/]
Nama Aliran Data	**Data Id Makanan**
Where used/how used	Proses 3.3
Deskripsi	Berisi data id makanan
Struktur data	id_makanan
id_makanan	[0-9]
Nama Aliran Data	**Data Tempat Makan**
Where used/how used	Proses 4, Proses 4.1, Proses 4.3, Proses 5.1, Proses 5.2
Deskripsi	Berisi Data Tempat Makan
Struktur data	id + nama + deskripsi + lokasi + jam_buka + jam_tutup + latitude + longitude + gambar_toko + *user*_id
id	[0-9]
nama	[A-Z|a-z]
deskripsi	[A-Z|a-z]
lokasi	[A-Z|a-z]
jam_buka	[0-9|:]
jam_tutup	[0-9|:]
latitude	[0-9]
longitude	[0-9]
gambar_toko	[0-9]
*user*_id	[0-9]

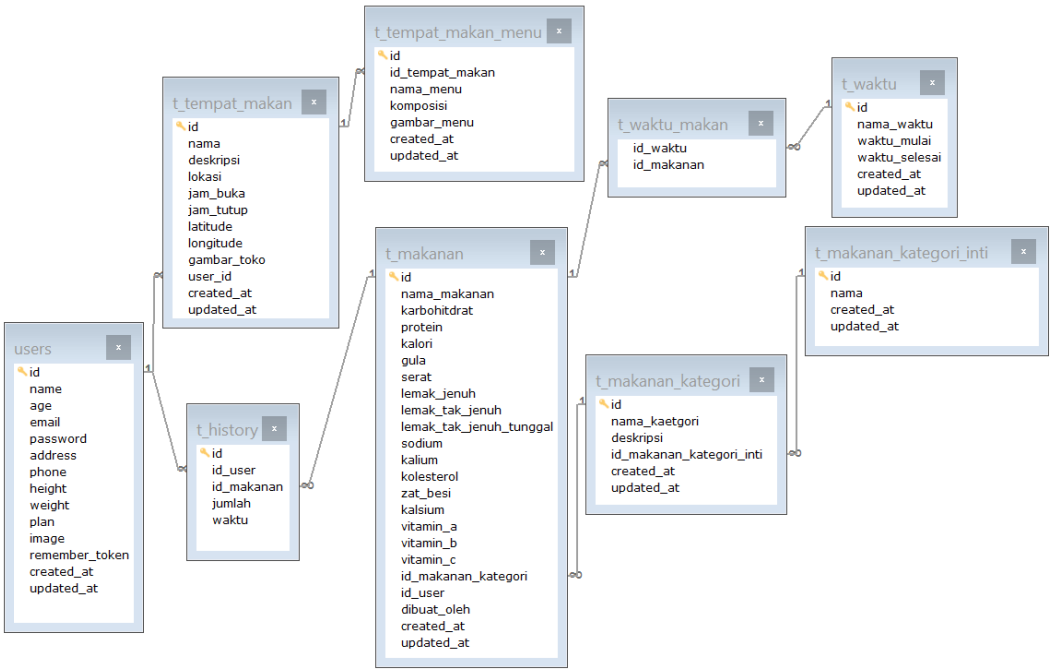
Nama Aliran Data	**Data Id Tempat Makan**
Where used/how used	Proses 4.1
Deskripsi	Bserisi data id tempat makan
Struktur data	id_tempat_makan
id_tempat_makan	[0-9]
Nama Aliran Data	**Data Menu Tempat Makan**
Where used/how used	Proses 5.0, Proses 5.1, Proses 5.2, Proses 5.3
Deskripsi	Berisi Data Menu Tempat Makan
Struktur data	id + id_tempat_makan + nama_menu + komposisi + gambar_menu
id	[0-9]
id_tempat_makan	[0-9]
nama_menu	[A-Z|a-z]
komposisi	[A-Z|a-z|0-9]
gambar_menu	[A-Z|a-z|0-9]

3.2. Perancangan Sistem

Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakuakn setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan sistem yang dimaksudkan untuk menggambarkan perbedaan antara sistem yang sedang berjalan dengan sistem yang diusulkan. Perancangan sistem ini menggunakan pendekatan sistem terkompsuterisasi.

3.2.1. Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Pada gambar dibawah ini, merupakan gambaran rangkaian basis data pada pembangunan aplikasi rekomendasi pola makan sehat dan rekomendasi tempat makan sehat.



Gambar 3.37 Skema Relasi

3.2.2. Struktur Tabel

Struktur tabel menggambarkan detail yang berisi field, tipe data, pangang data, dan keterangan lainnya. Tabel yang terdapat pada basis data pembangunan aplikasi untuk rekomendasi pola makan dan tempat makan sehat memanfaatkan API *clarifai* dan teknologi GPS pada *platform andorid* ini adalah t_*user*, t_tempat_makan, t_makan_menu, t_history, t_makanan, t_makanan_kategori, t_waktu_makan, t_waktu, t_makanan_kategori_inti.

1. Tabel t_*user*

   Berikut adalah Tabel 3.27 Struktur Tabel t_*user* yang digunakan pada basis data perangkat lunak.

Tabel 3.27 Struktur Tabel t_user

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	10	Tidak	auto_increment	Primary Key
name	Varchar	255	Tidak		
age	Int	3	Ya		
email	Varchar	255	Tidak		
password	Varchar	60	Tidak		
address	*Text*		Ya		
phone	Decimal	15,0	Ya		
height	Int	3	Ya		
weight	Int	3	Ya		

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
plan	Varchar	100	Ya		
image	Varchar	30	Ya		
remember_token	Varchar	100	Ya		
created_at	Timestamp		Tidak		
update_at	Timestamp		Tidak		

2. Tabel t_tempat_makan

Berikut adalah Tabel 3.28 Struktur Tabel t_tempat_makan yang digunakan pada basis data perangkat lunak.

Tabel 3.28 Struktur Tabel t_tempat_makan

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	10	Tidak	auto_increment	Primary Key
nama	Varchar	100	Ya		
deskripsi	*Text*		Ya		
lokasi	*Text*		Ya		
jam_buka	Time		Ya		
Jam_tutup	Time		Ya		
latitude	Double		Ya		
longitude	Double		Ya		
gambar_toko	Varchar	30	Ya		
*user*_id	Int	10	Ya		Foreign Key
created_at	Datetime		Ya		
update_at	Datetime		Ya		

3. Tabel t _makan_menu

Berikut adalah Tabel 3.29 Struktur Tabel t_makan_menu yang digunakan pada basis data perangkat lunak.

Tabel 3.29 Struktur Tabel t_makan_menu

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	10	Tidak	auto_increment	Primary Key
id_tempat_makan	Int	10	Ya		Foreign Key
nama_menu	Varchar	100	Ya		
komposisi	*Text*		Ya		
gambar_menu	Varchar	100	Ya		
created_at	Datetime		Ya		
update_at	Datetime		Ya		

4. Tabel t_history

Berikut adalah Tabel 3.30 Struktur Tabel t_history yang digunakan pada basis data perangkat lunak.

Tabel 3.30 Struktur Tabel t_history

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	11	Tidak	auto_Increment	Primary Key
id_*user*	Int	11	Tidak		Foreign Key
id_makanan	Int	11	Tidak		Foreign Key
jumlah	Int	11	Tidak		
waktu	Datetime		Tidak		

5. Tabel t_makanan

Berikut adalah Tabel 3.31 Struktur Tabel t_makanan yang digunakan pada basis data perangkat lunak.

Tabel 3.31 Struktur Tabel t_makanan

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	11	Tidak	auto_increment	Primary Key
nama_makanan	Varchar	100	Ya		
karbohidrat	Double		Ya		
protein	Double		Ya		
kalori	Double		Ya		
gula	Double		Ya		
serat	Double		Ya		
lemak_jenuh	Double		Ya		
lemak_tak_jenuh	Double		Ya		
lemak_tak_jenuh_tunggal	Double		Ya		
sodium	Double		Ya		
kalium	Double		Ya		
kolesterol	Double		Ya		
zatbesi	Double		Ya		
kalsium	Double		Ya		
vitamin_a	Double		Ya		
vitamin_b	Double		Ya		
vitamin_c	Double		Ya		
id_makanan_kategori	Int	10	Ya		Foreign Key
id_*user*	Int	10	Tidak		
dibuat_oleh	Varchar	20	Tidak		
created_at	Datetime		Tidak		
update_at	Datetime		Ya		

6. Tabel t_makanan_kategori

Berikut adalah Tabel 3.32 Struktur Tabel t_makanan_kategori yang digunakan pada basis data perangkat lunak.

Tabel 3.32 Strutur Tabel t_makanan_kategori

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	11	Tidak	auto_increment	Primary Key
nama_kategori	Varchar	100	Tidak		
deskripsi	*Text*		Ya		
id_makanan_kategori_inti	Int	11	Tidak		Foreign Key
created_at	Datetime		Tidak		

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
update_at	Datetime		Ya		

7. Tabel t_waktu_makan

Berikut adalah Tabel 3.33 Struktur Tabel t_waktu_makan yang digunakan pada basis data perangkat lunak.

Tabel 3.33 Struktur Tabel t_waktu_makan

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_waktu	Int	11	Tidak		Foreign Key
id_makanan	Int	11	Tidak		Foreign Key

8. Tabel t_waktu

Berikut adalah Tabel 3.34 Struktur Tabel t_waktu yang digunakan pada basis data perangkat lunak.

Tabel 3.34 Struktur Tabel t_waktu

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	10	Tidak	auto_increment	Primary Key
nama_waktu	Varchar	100	Tidak		
waktu_mulai	Time		Ya		
waktu_selesai	Time		Ya		
created_at	Datetime		Ya		
update_at	Datetime		Ya		

9. Tabel t_makanan_kategori_inti

Berikut adalah Tabel 3.35 Struktur Tabel t_makanan_kategori_inti yang digunakan pada basis data perangkat lunak.

Tabel 3.35 Struktur Tabel t_makanan_kategori_inti

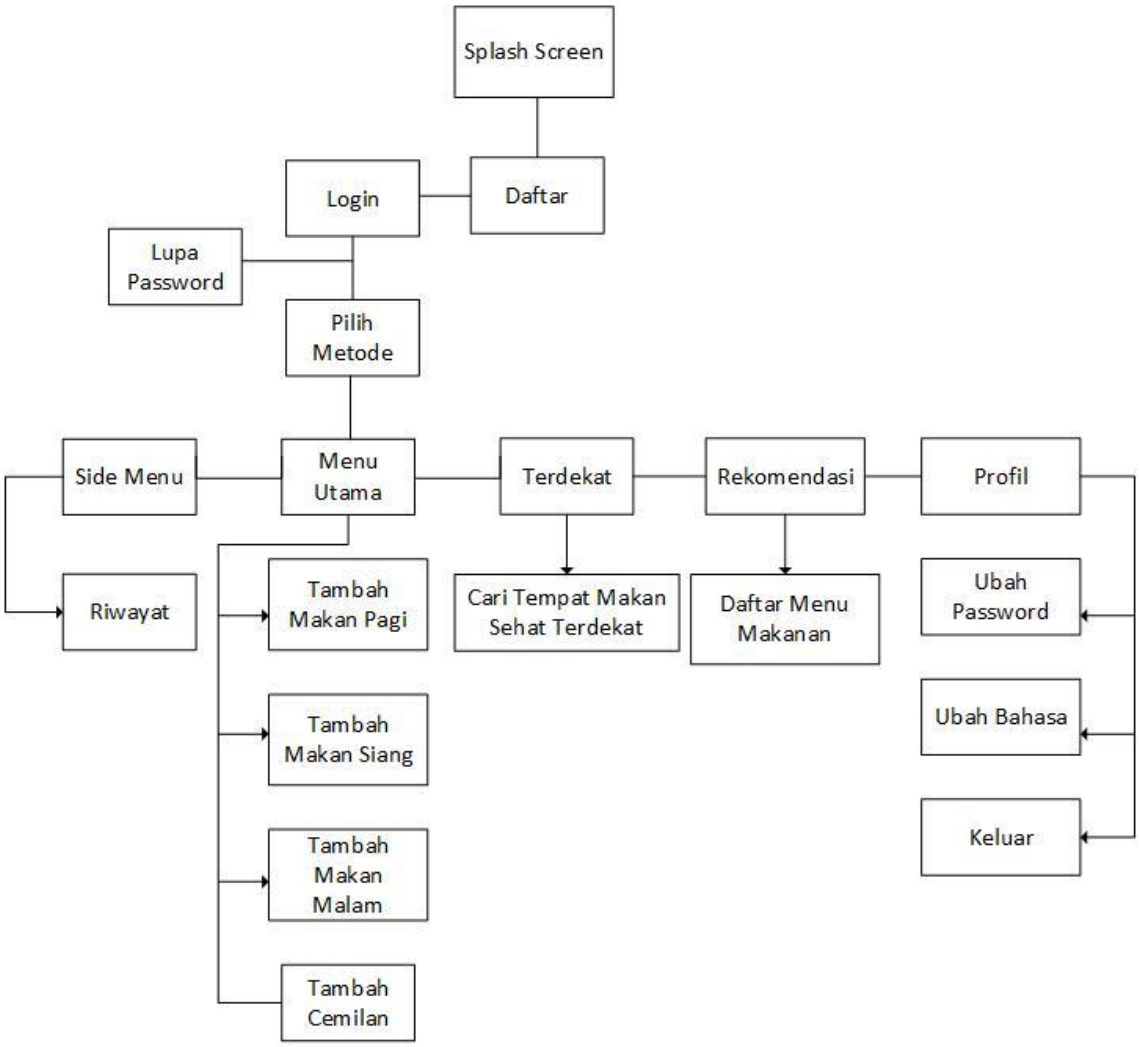
Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id	Int	11	Tidak	auto_increment	Primary Key
nama	Varchar	100	Tidak		
created_at	Datetime		Tidak		
update_at	Datetime		Ya		

3.2.3. Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi sehingga aplikasi yang mudah dipahami dan mudah digunakan. Perancangan struktur menu dari pembangunan aplikasi untuk rekomendasi pola makan dan tempat makan sehat memanfaatkan API *clarifai* dan teknologi GPS pada *platform andorid* dapat dilihat sebagai berikut :

1. Struktur Menu *Platform Mobile*

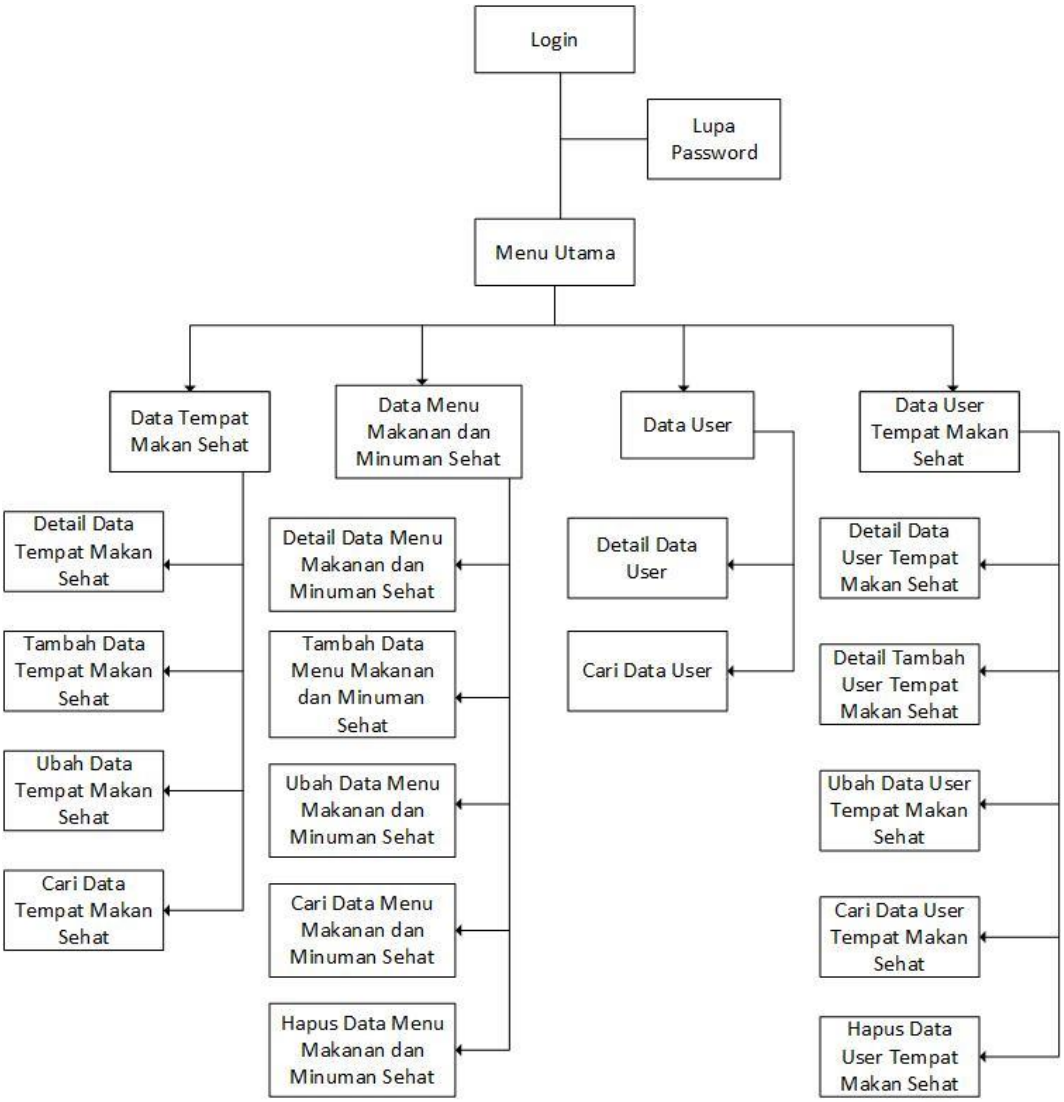
   Berikut ini adalah Gambar 3.38 Struktur Menu Platfor *Mobile*



Gambar 3.38 Struktur Menu *Platform Mobile*

2. Struktur Menu *Platform Web Admin*

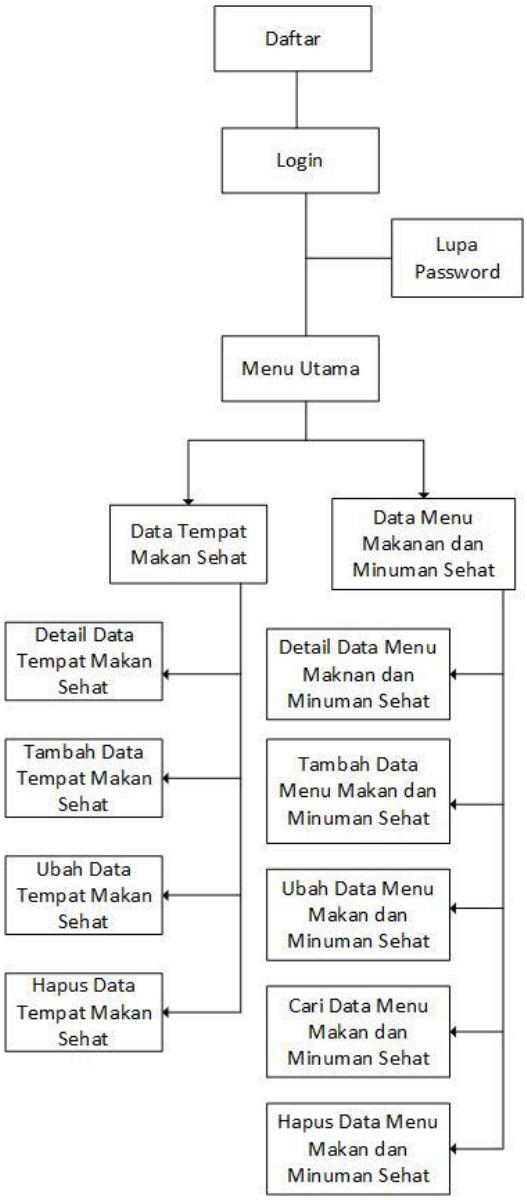
Berikut adalah Gambar 3.39 Struktur Menu *Platform Web Admin*



Gambar 3.39 Struktur Menu *Web Admin*

3. Struktur Menu *Web User* Tempat Makan Sehat

Berikut adalah Gambar 3.40 Struktur Menu *Platform Web User* Tempat Makan Sehat



Gambar 3.40 Struktur Menu *Web User* Tempat Makan Sehat

3.2.4. Perancangan Antarmuka

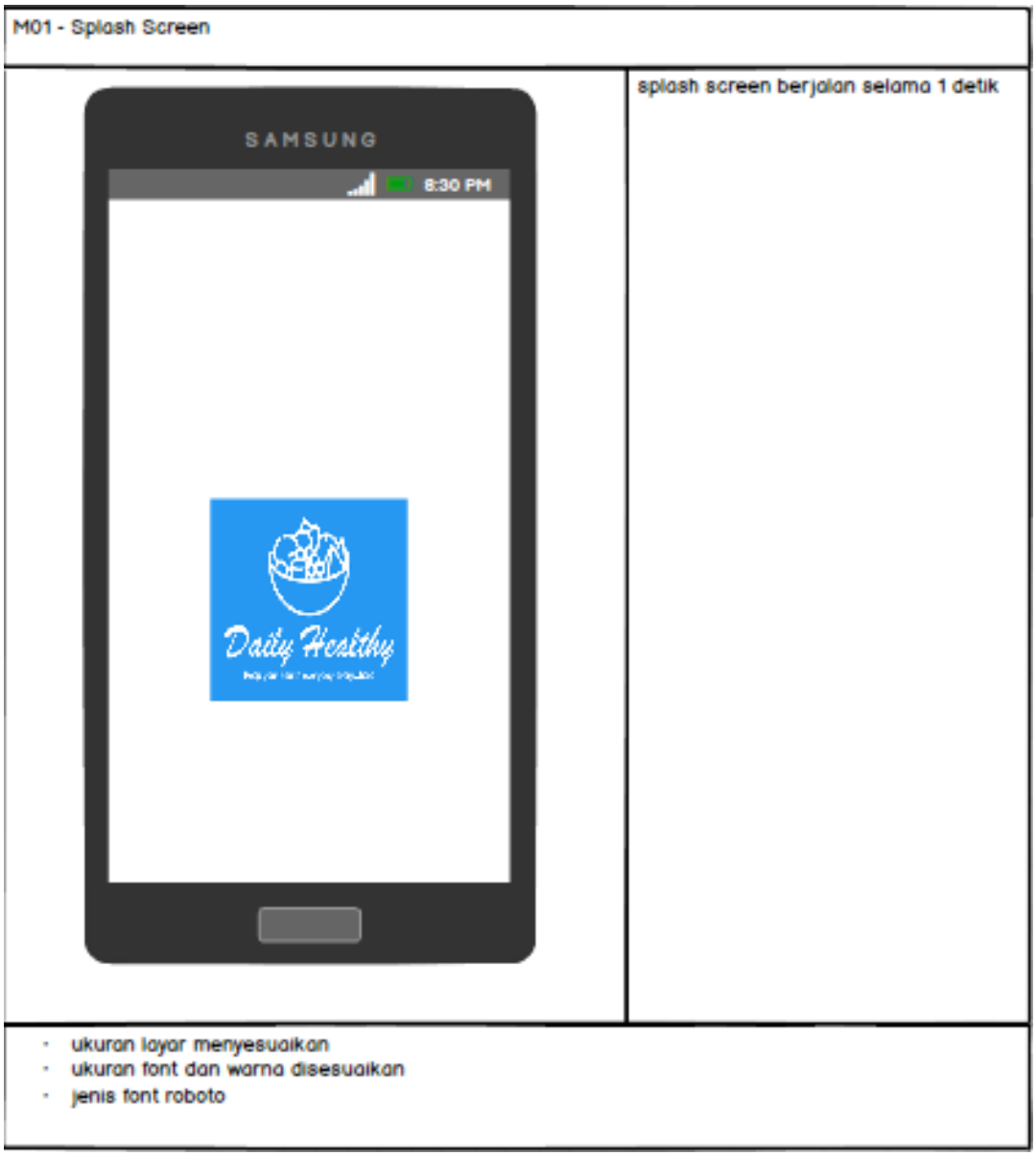
Perancangan antarmuka mendeskripsikan rencana tampilan dari setiap tampilan yang akan digunakan pada pembangunan aplikasi rekomendasi pola makan sehat dan rekomendasi tempat makan sehat terdiri dari perancangan form dan pesan.

3.2.4.1. Perancangan Antarmuka Pada *Platform Mobile*

Perancangan antarmuka pada *Platform Mobile* pada pembangunan aplikasi rekomendasi pola makan sehat dan rekomendasi tempat makan sehat dapat dilihat pada gambar-gambar dibawah ini.

1. Perancangan Antarmuka *Splash screen*

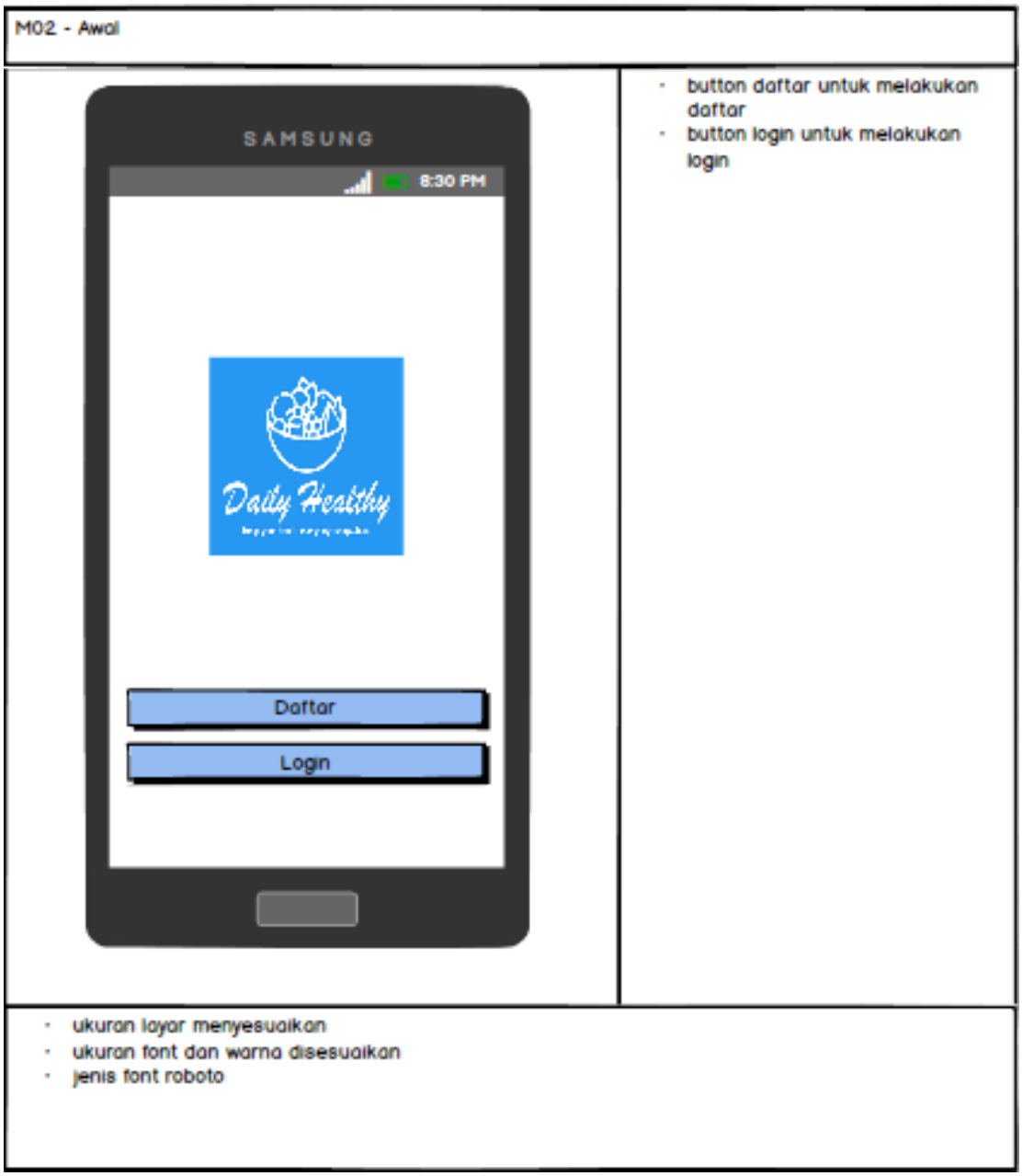
Berikut ini merupakan antarmuka *splash screen* pada *platform mobile* yang dapat dilihat pada Gambar 3.41 Perancangan Antarmuka *Splash screen*.



Gambar 3.41 Perancangan Antarmuka *Splash screen*

2. Perancangan Antarmuka Awal

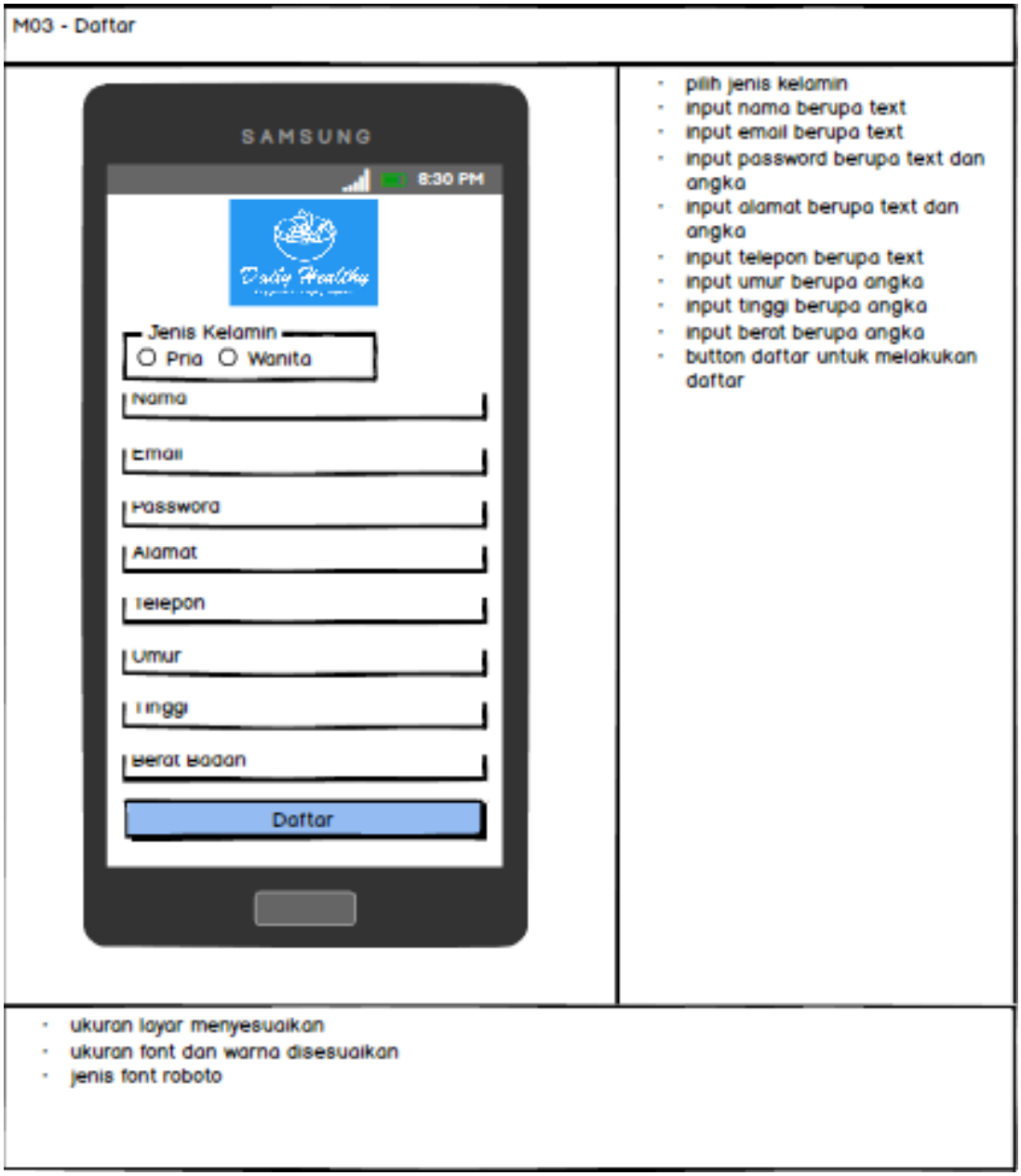
Berikut ini merupakan antarmuka awal pada *platform mobile* yang dapat dilihat pada Gambar 3.42 Perancangan Antarmuka Awal.



Gambar 3.42 Perancangan Antarmuka Awal

3. Perancangan Antarmuka Daftar

Berikut ini merupakan antarmuka daftar pada *platform mobile* yang dapat dilihat pada Gambar 3.43 Perancangan Antarmuka Daftar.



Gambar 3.43 Perancangan Antarmuka Daftar

4. Perancangan Antarmuka Pilih Metode

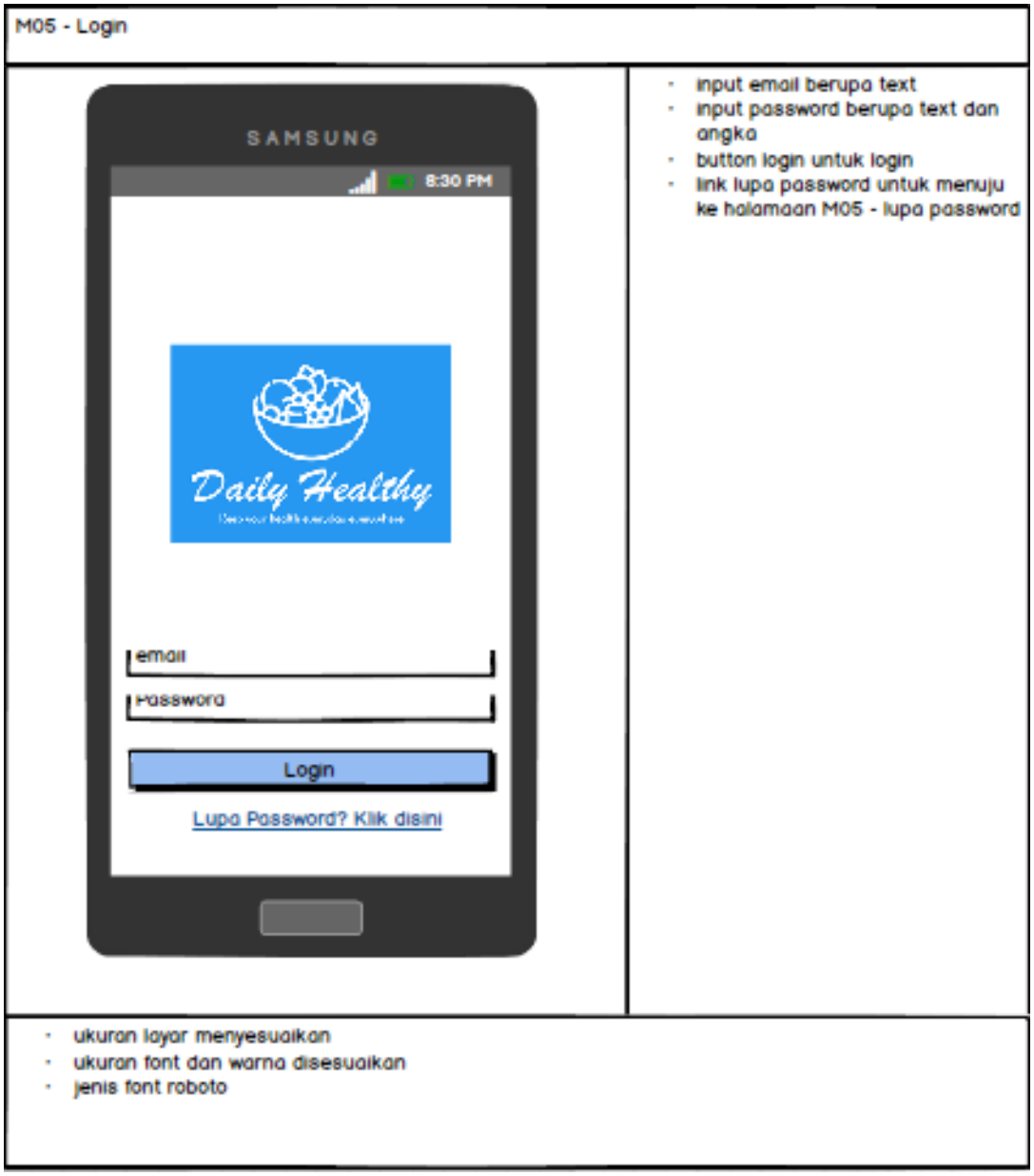
Berikut ini merupakan antarmuka pilih metode pada *platform mobile* yang dapat dilihat pada Gambar 3.44 Perancangan Antarmuka Pilih Metode.



Gambar 3.44 Perancangan Antarmuka Pilih Metode

5. Perancangan Antarmuka *Login*

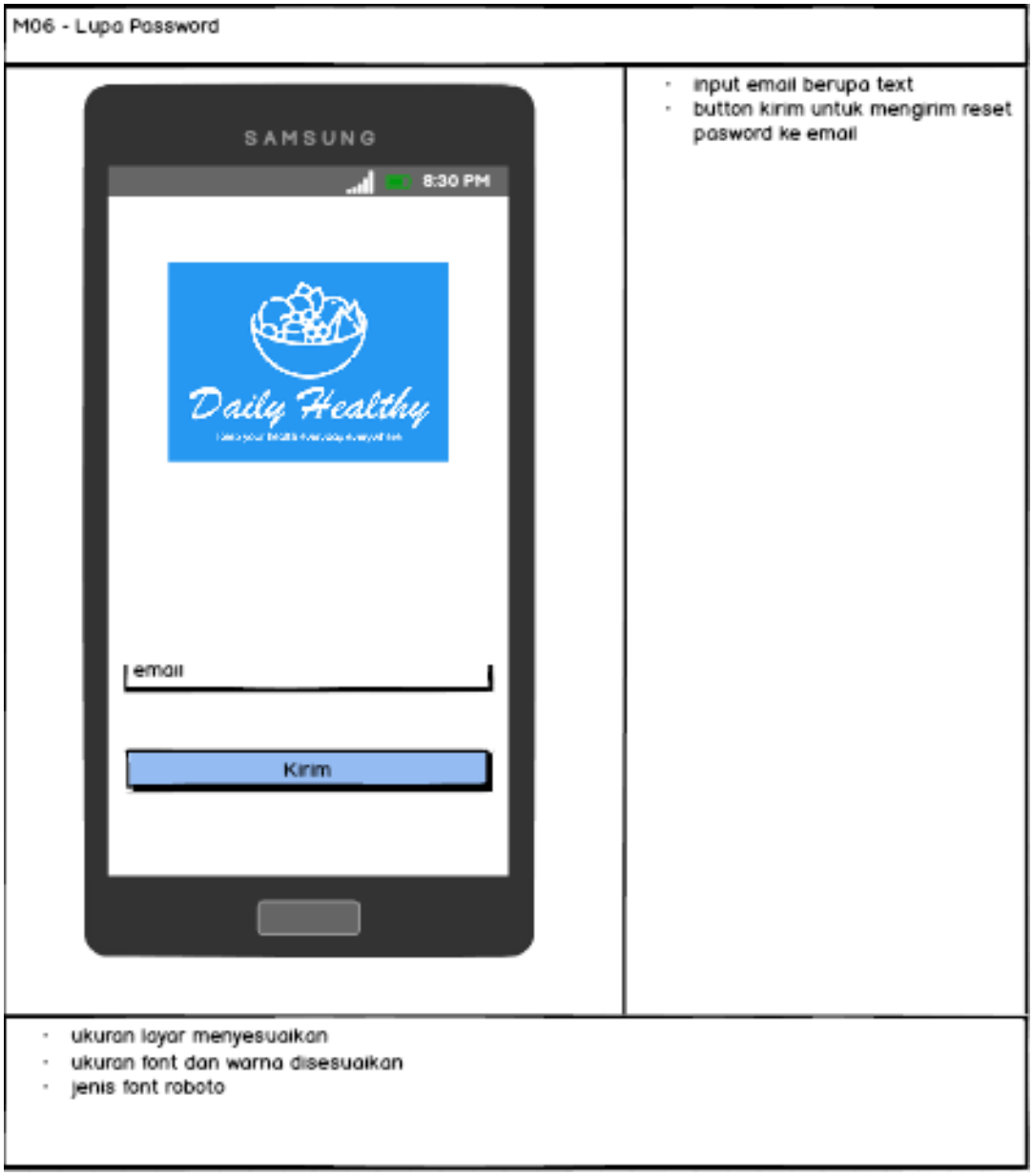
Berikut ini merupakan antarmuka *login* pada *platform mobile* yang dapat dilihat pada Gambar 3.45 Perancangan Antarmuka *Login*.



Gambar 3.45 Perancangan Antarmuka Login

6. Perancangan Antarmuka Lupa *Password*

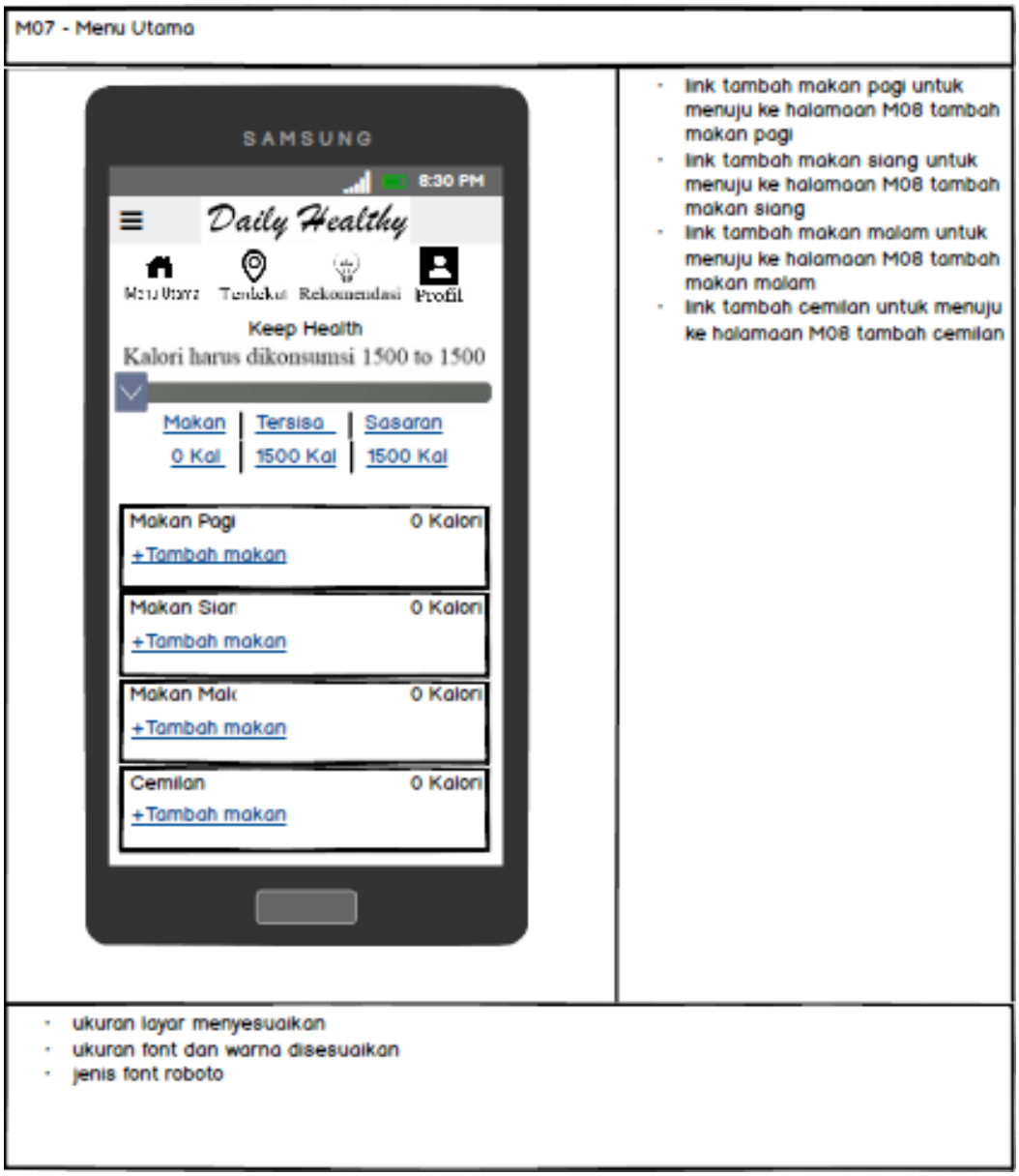
Berikut ini merupakan antarmuka lupa *password* pada *platform mobile* yang dapat dilihat pada Gambar 3.46 Perancangan Antarmuka Lupa *Password*.



Gambar 3.46 Perancangan Antarmuka Lupa Password

7. Perancangan Antarmuka Menu Utama

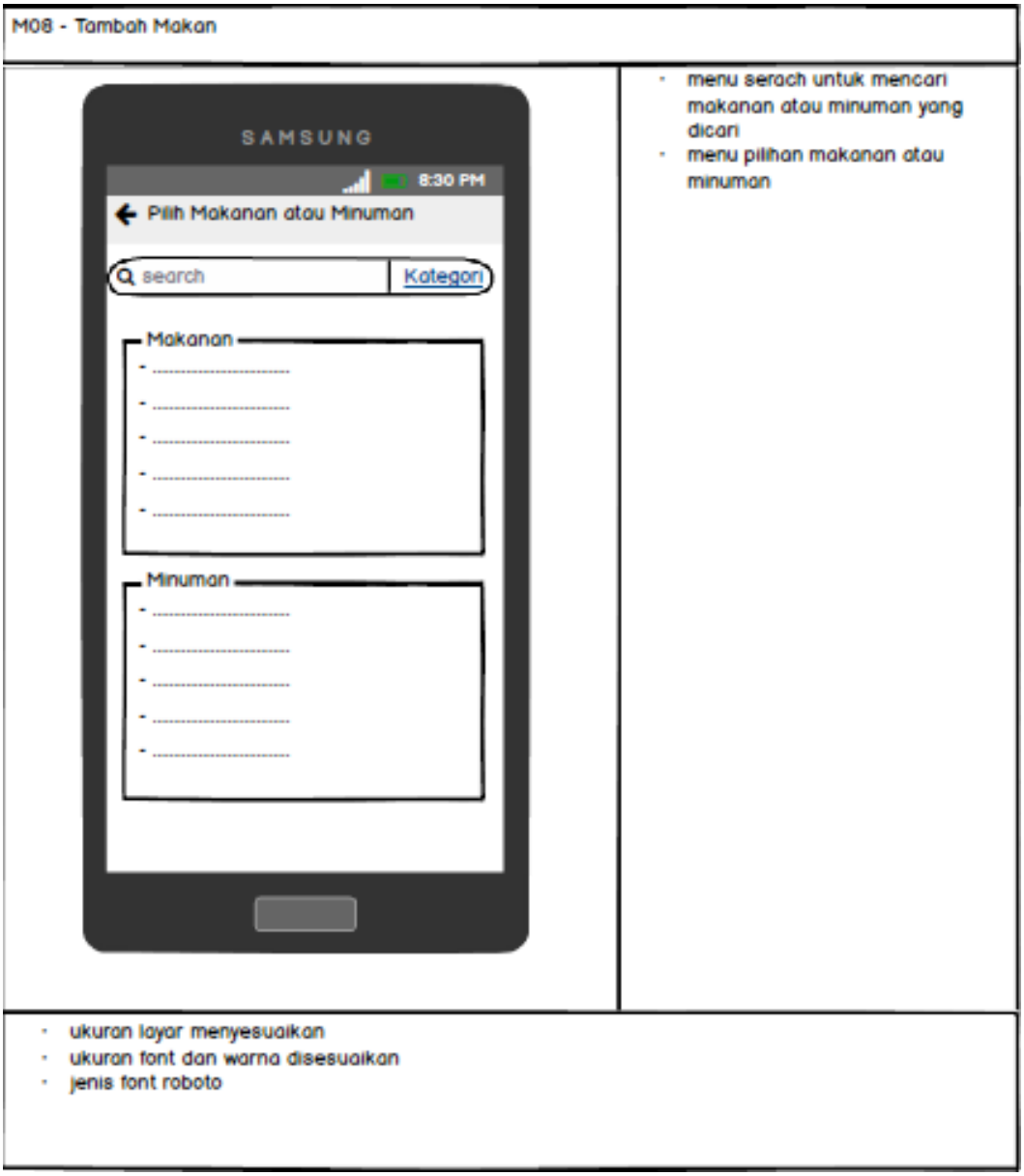
Berikut ini merupakan antarmuka menu utama pada *platform mobile* yang dapat dilihat pada Gambar 3.47 Perancangan Antarmuka Menu Utama.



Gambar 3.47 Perancangan Antarmuka Menu Utama

8. Perancangan Antarmuka Tambah Makan

Berikut ini merupakan antarmuka tambah makan pada *platform mobile* yang dapat dilihat pada Gambar 3.48 Perancangan Antarmuka Tambah Makan.



Gambar 3.48 Perancangan Antarmuka Tambah Makan

9. Perancangan Antarmuka Menu Terdekat

Berikut ini merupakan antarmuka menu terdekat pada *platform mobile* yang dapat dilihat pada Gambar 3.49 Perancangan Antarmuka Menu Terdekat.



Gambar 3.49 Perancangan Antarmuka Menu Terdekat

10. Perancangan Antarmuka Menu Rekomendasi

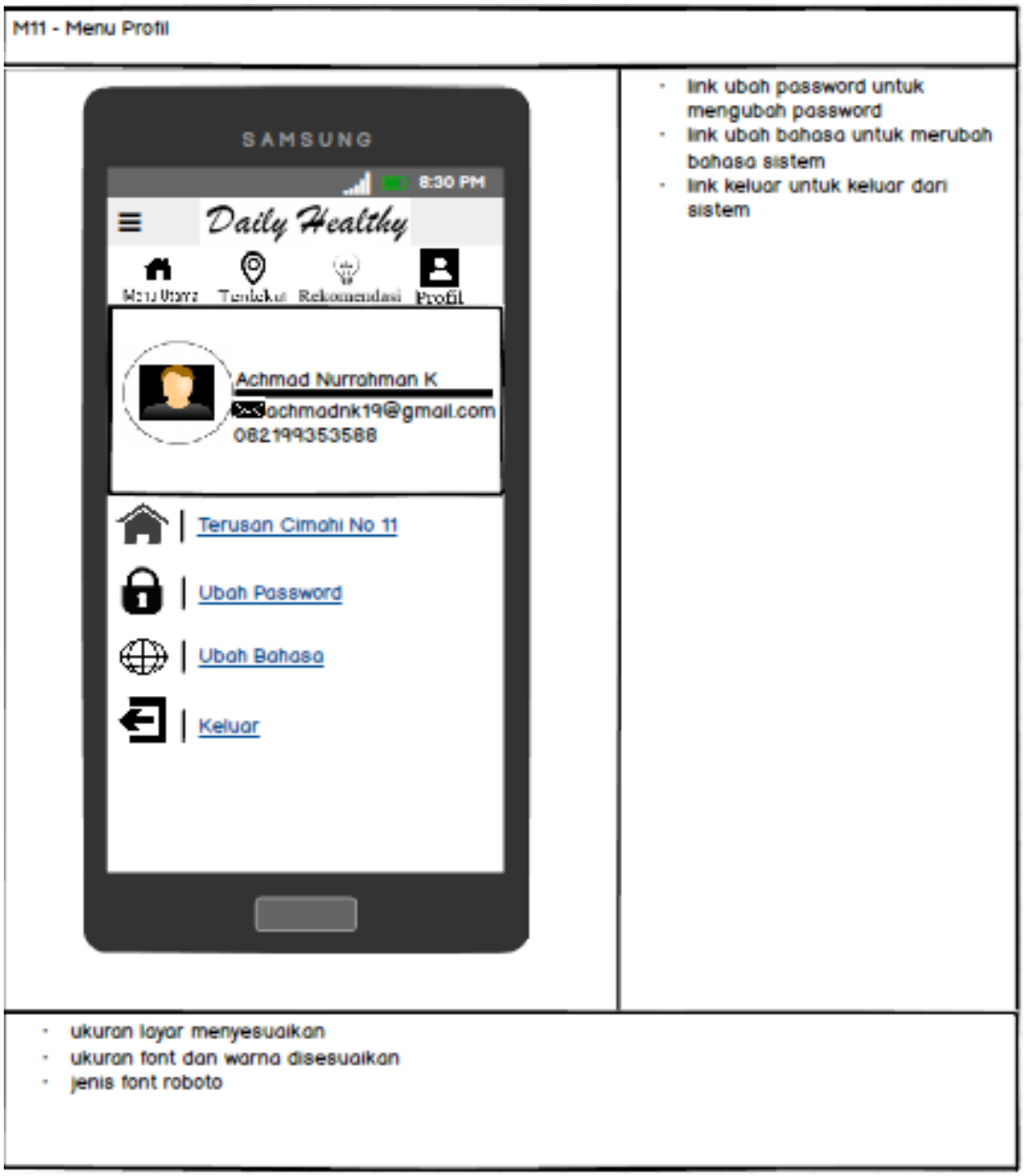
Berikut ini merupakan antarmuka menu rekomendasi pada *platform mobile* yang dapat dilihat pada Gambar 3.50 Perancangan Antarmuka Menu Rekomendasi.



Gambar 3.50 Perancangan Antarmuka Menu Rekomendasi

11. Perancangan Antarmuka Profil

Berikut ini merupakan antarmuka menu *profil* pada *platform mobile* yang dapat dilihat pada Gambar 3.51 Perancangan Antarmuka Menu Profil.



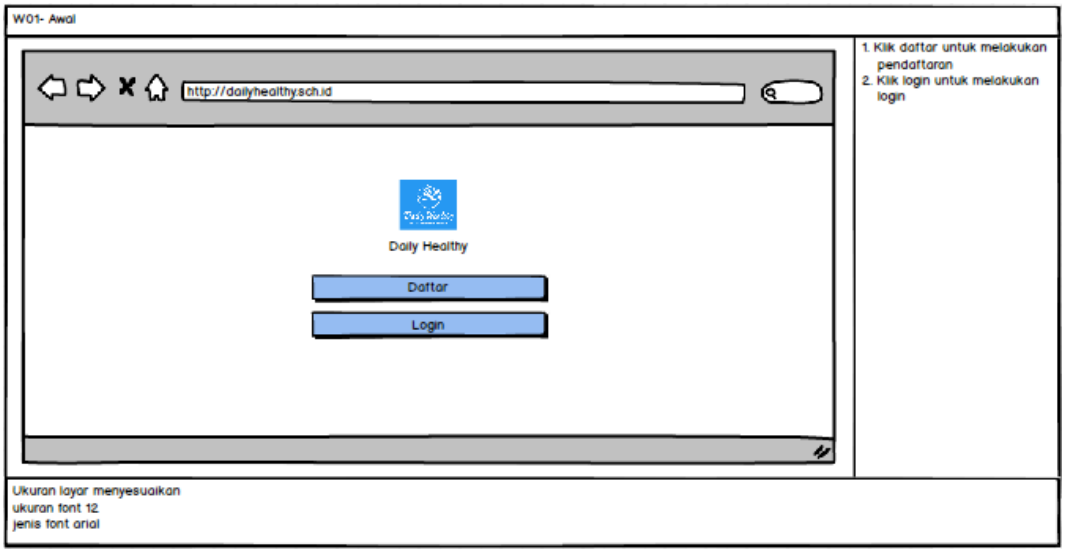
Gambar 3.51 Perancangan Antarmuka Menu Profil

3.2.4.2. Perancangan Antarmuka Pada *Platform Web Admin*

Perancangan antarmuka pada *Platform WEB Admin* pada pembangunan aplikasi rekomendasi pola makan sehat dan rekomendasi tempat makan sehat dapat dilihat pada gambar-gambar dibawah ini.

1. Perancangan Antarmuka Awal

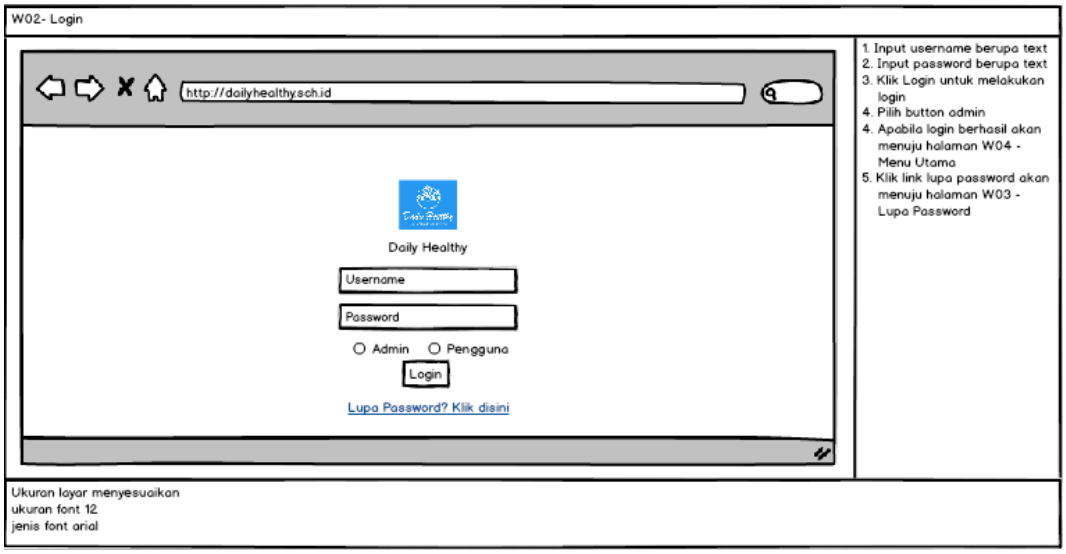
   Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.52 Perancangan Antarmuka Awal.



Gambar 3.52 Perancangan Antarmuka Awal

2. Perancangan Antarmuka *Login*

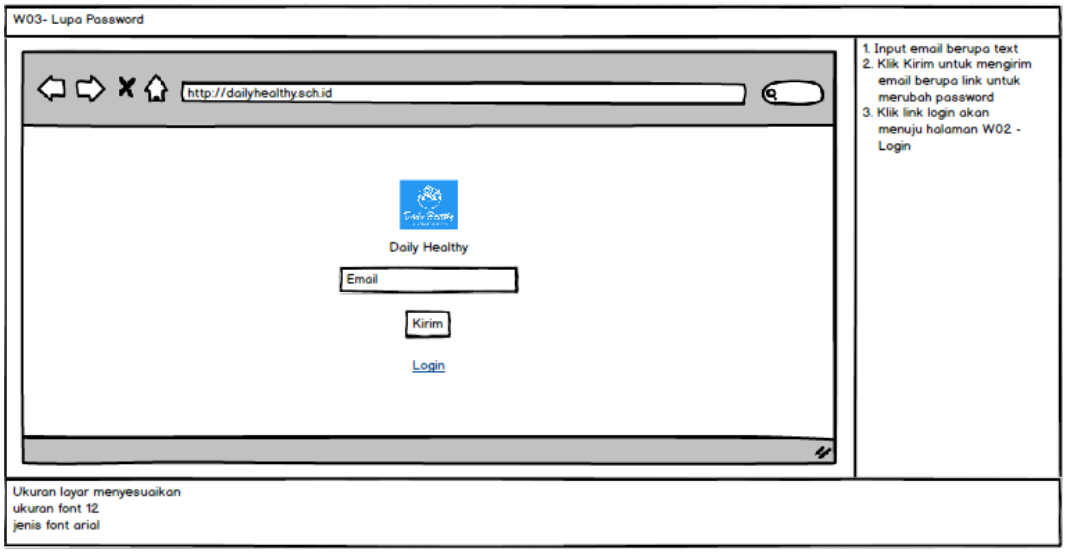
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.53 Perancangan Antarmuka *Login*.



Gambar 3.53 Perancangan Antarmuka Login

3. Perancangan Antarmuka Lupa *Password*

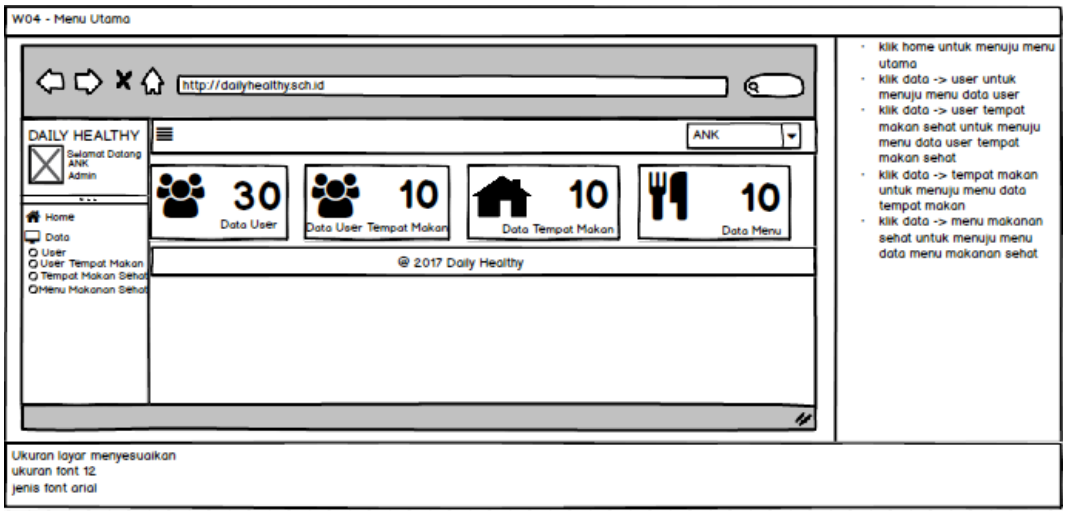
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.54 Perancangan Antarmuka Lupa *Password*.



Gambar 3.54 Perancangan Antarmuka Lupa Password

4. Perancangan Antarmuka Menu Utama

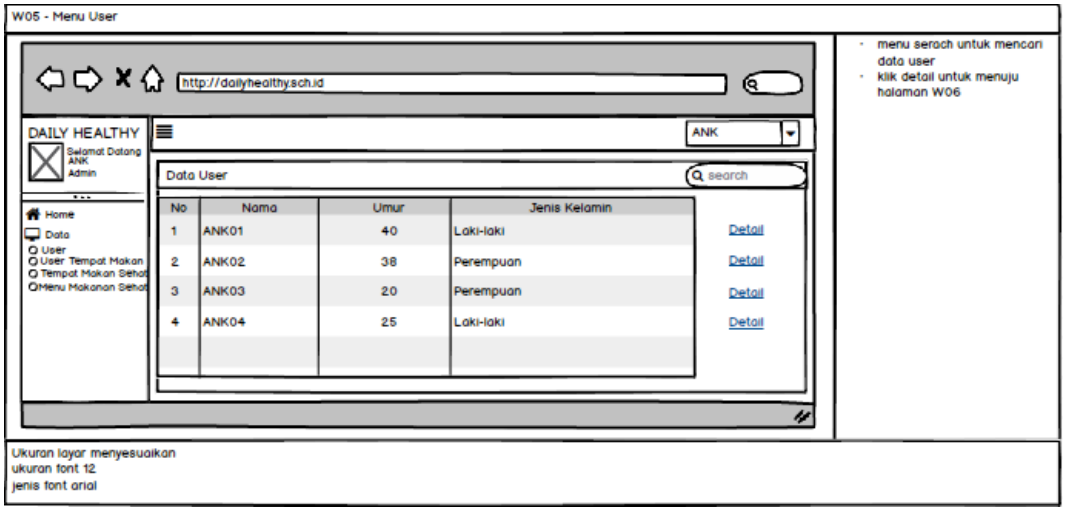
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.55 Perancangan Antarmuka Menu Utama.



Gambar 3.55 Perancangan Antarmuka Menu Utama

5. Perancangan Antarmuka Menu Data *User*

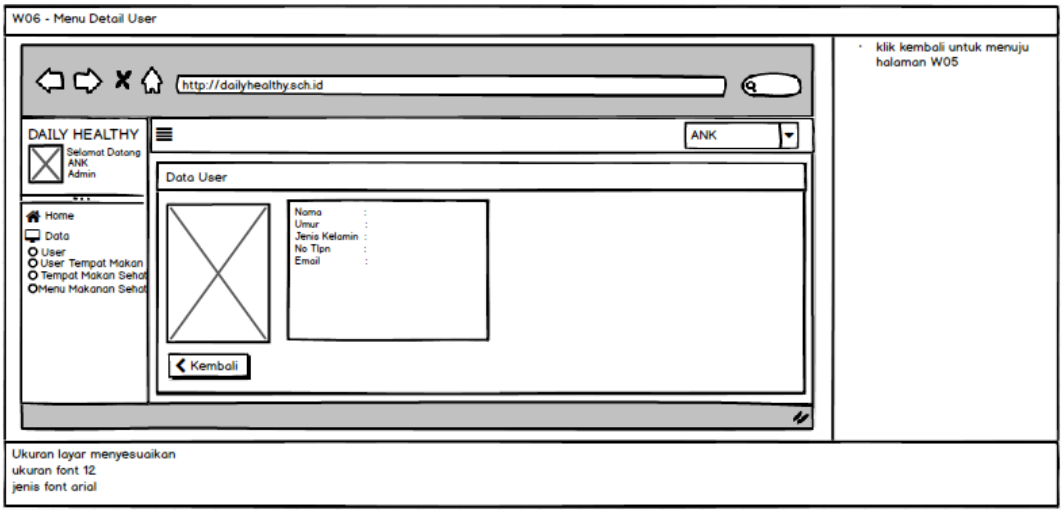
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.56 Perancangan Antarmuka Menu Data *User*.



Gambar 3.56 Perancangan Antarmuka Menu Data User

6. Perancangan Antarmuka Menu Data Detail *User*

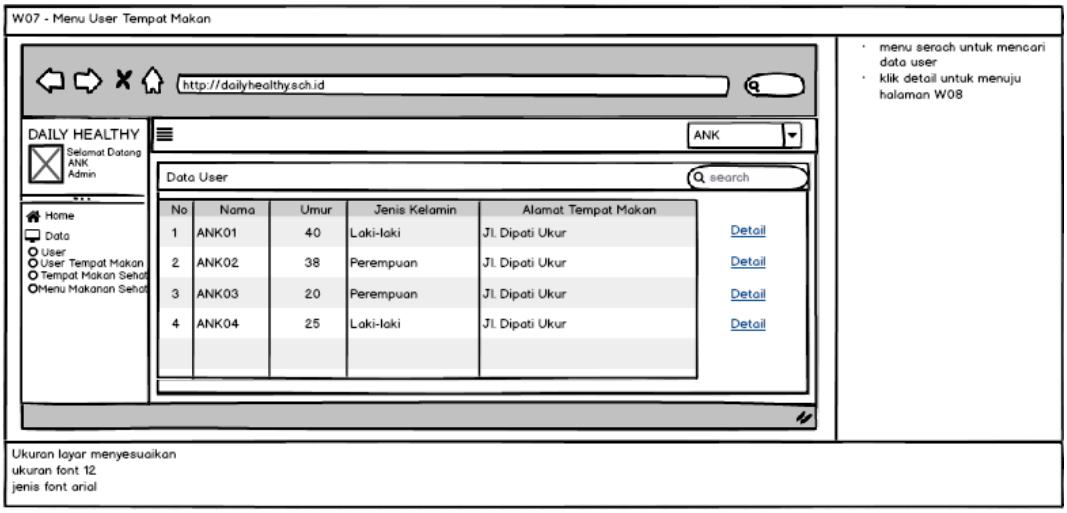
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.57 Perancangan Antarmuka Menu Data Detail *User*.



Gambar 3.57 Perancangan Antarmuka Menu Data Detail User

7. Perancangan Antarmuka Menu Data *User* Tempat Makan

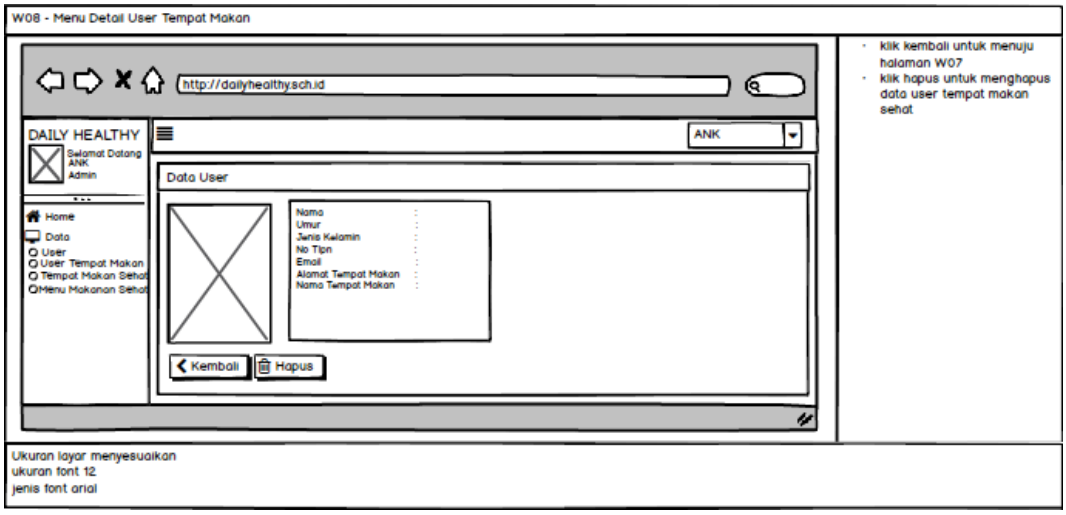
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.58 Perancangan Antarmuka Menu Data *User* Tempat Makan.



Gambar 3.58 Perancangan Antarmuka Menu Data User Tempat Makan

8. Perancangan Antarmuka Menu Data Detail *User* Tempat Makan

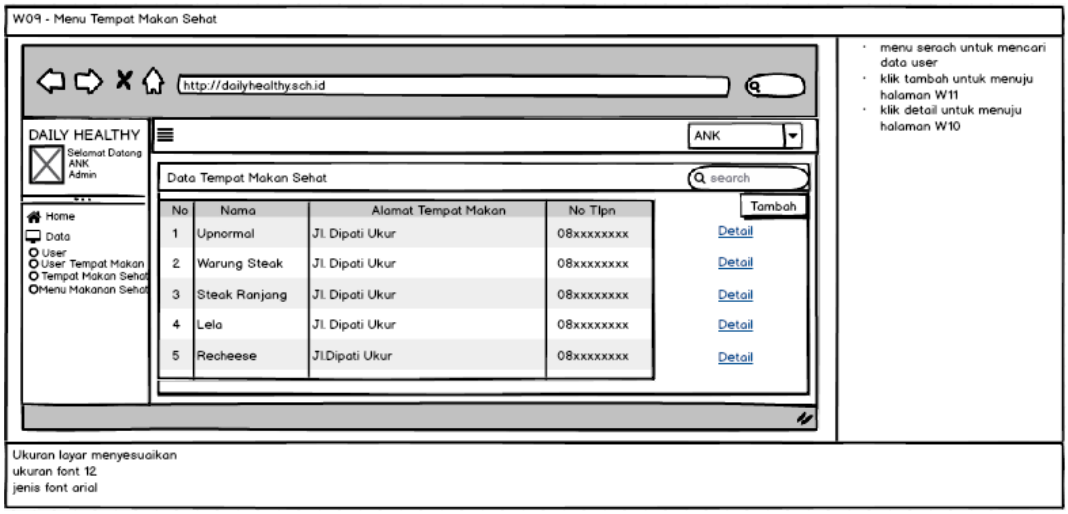
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.59 Perancangan Antarmuka Menu Data Detail *User* Tempat Makan.



Gambar 3.59 Perancangan Antarmuka Menu Data Detail User Tempat Makan

9. Perancangan Antarmuka Menu Data Tempat Makan Sehat

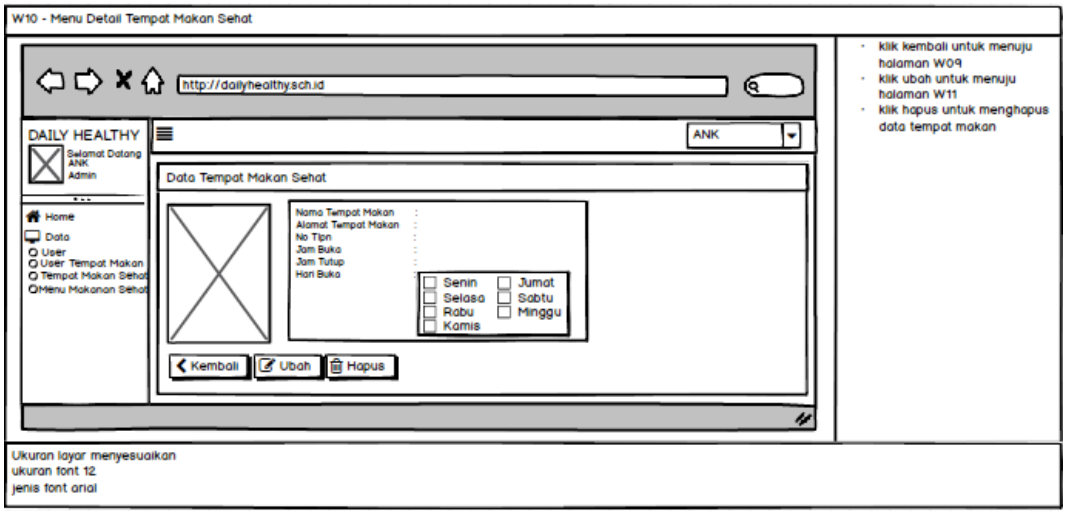
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.60 Perancangan Antarmuka Menu Data Tempat Makan Sehat.



Gambar 3.60 Perancangan Antarmuka Menu Data Tempat Makan Sehat

10. Perancangan Antarmuka Menu Data Detail Data Tempat Makan Sehat

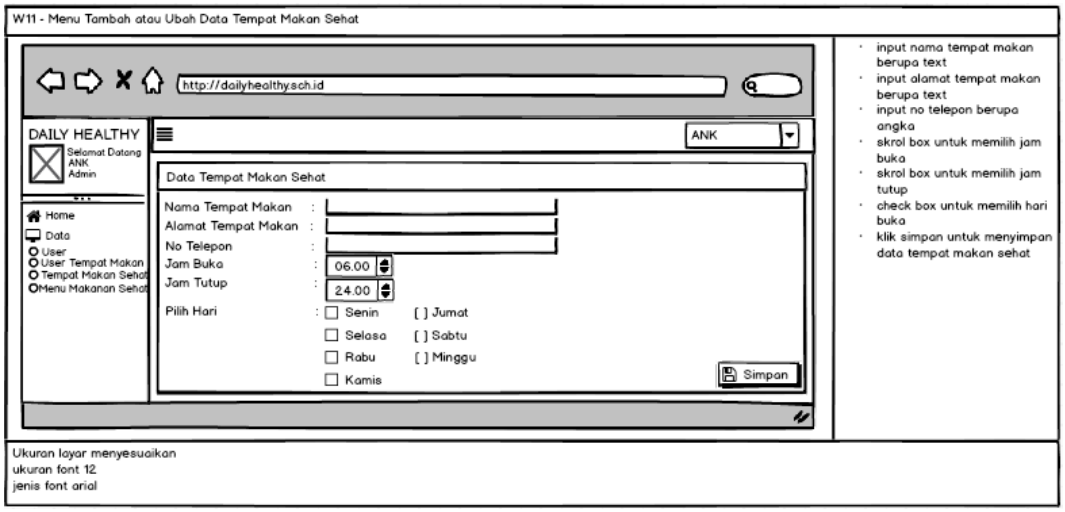
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.61 Perancangan Antarmuka Menu Data Detail Data Tempat Makan Sehat.



Gambar 3.61 Perancangan Antarmuka Menu Data Detail Data Tempat Makan Sehat

11. Perancangan Antarmuka Menu Tambah atau Ubah Data Tempat Makan Sehat

Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.62 Perancangan Antarmuka Menu Tambah atau Ubah Data Tempat Makan Sehat.



Gambar 3.62 Perancangan Antarmuka Menu Tambah atau Ubah Data Tempat Makan Sehat

12. Perancangan Antarmuka Menu Data Makanan Sehat

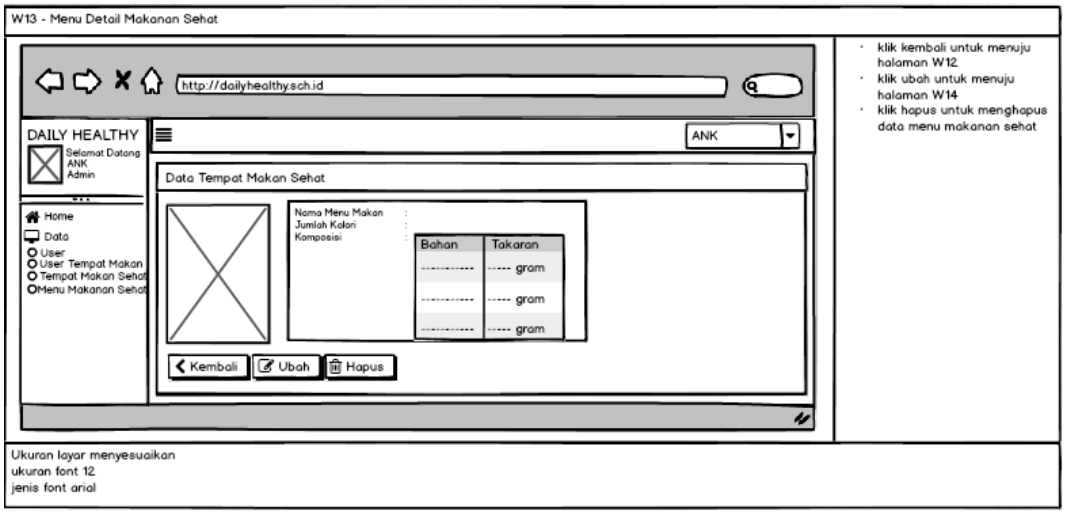
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.63 Perancangan Antarmuka Menu Data Menu Makanan Sehat.



Gambar 3.63 Perancangan Antarmuka Menu Data Menu Makanan Sehat

13. Perancangan Antarmuka Menu Data Detail Makanan Sehat

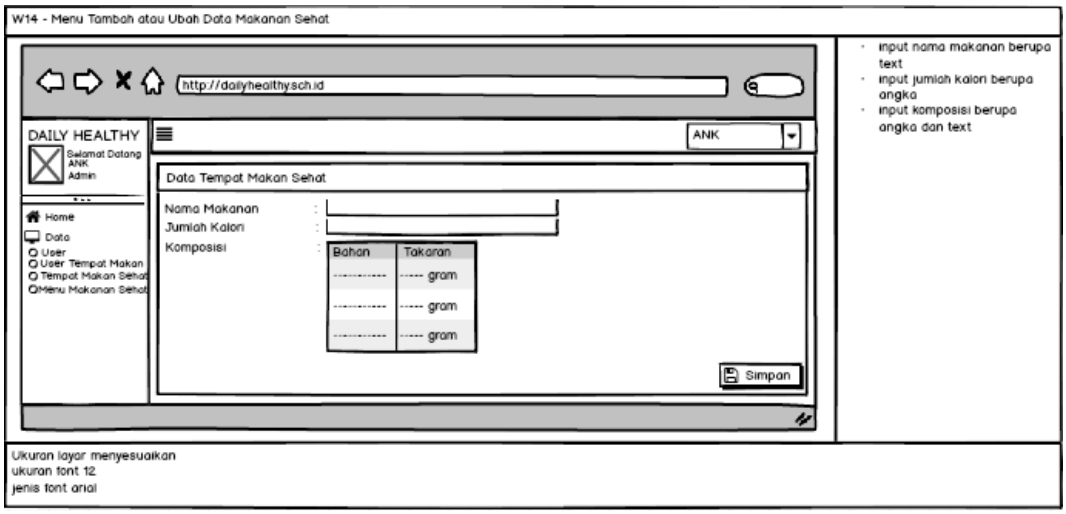
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.64 Perancangan Antarmuka Menu Data Detail Makanan Sehat.



Gambar 3.64 Perancangan Antarmuka Menu Data Detail Makanan Sehat

14. Perancangan Antarmuka Menu Tambah atau Ubah Data Makanan Sehat

Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.65Perancangan Antarmuka Menu Tambah atau Ubah Data Makanan Sehat.



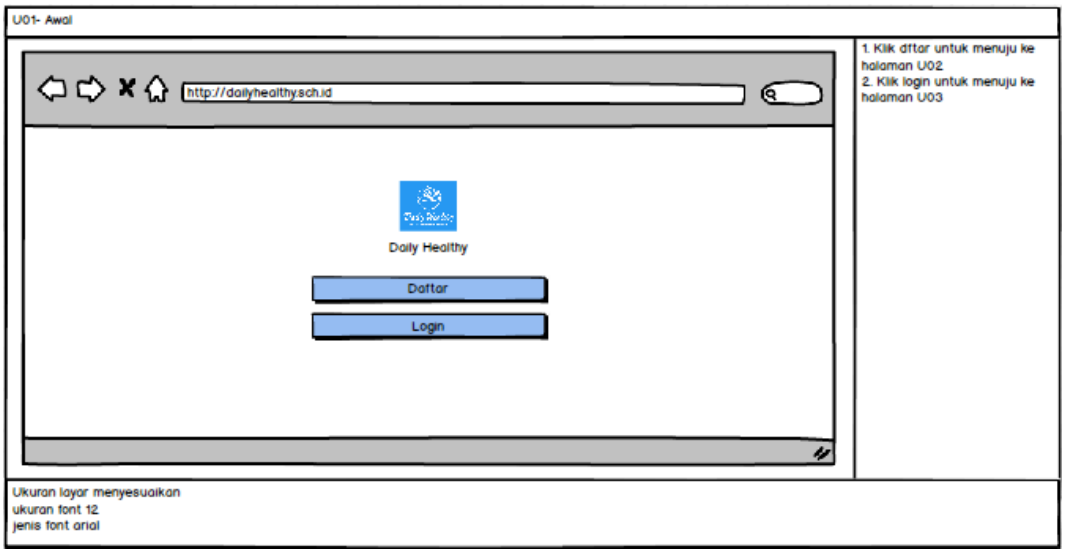
Gambar 3.65Perancangan Antarmuka Menu Tambah atau Ubah Data Makanan Sehat

3.2.4.3. Perancangan Antarmuka Pada Platform Web User

Perancangan antarmuka pada *Platform WEB User* Tempat Makan Sehat pada pembangunan aplikasi rekomendasi pola makan sehat dan rekomendasi tempat makan sehat dapat dilihat pada gambar-gambar dibawah ini.

1. Perancangan Antarmuka Awal

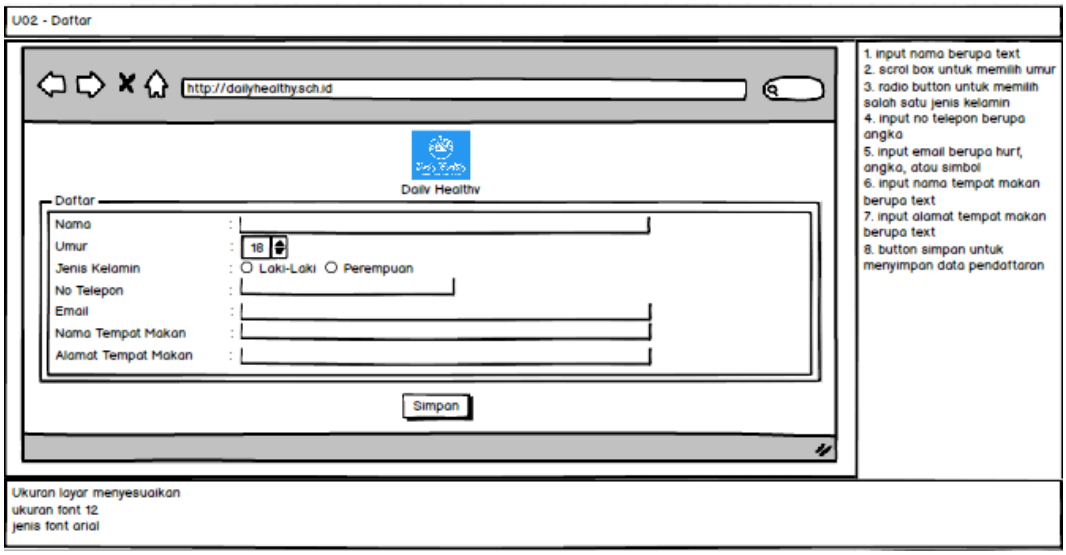
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.66 Perancangan Antarmuka Awal.



Gambar 3.66 Perancangan Antarmuka Awal

2. Perancangan Antarmuka Daftar

Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.67 Perancangan Antarmuka Daftar.



Gambar 3.67 Perancangan Antarmuka Daftar

3. Perancangan Antarmuka *Login*

Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.68 Perancangan Antarmuka *Login*.



Gambar 3.68 Perancangan Antarmuka Login

4. Perancangan Antarmuka Lupa *Password*

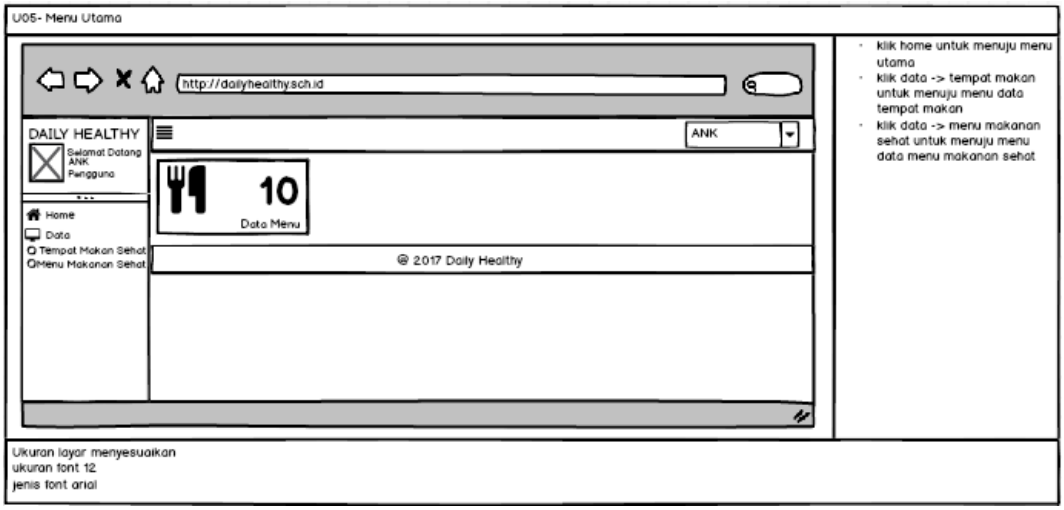
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.69 Perancangan Antarmuka Lupa *Password*.



Gambar 3.69 Perancangan Antarmuka Lupa Password

5. Perancangan Antarmuka Menu Utama

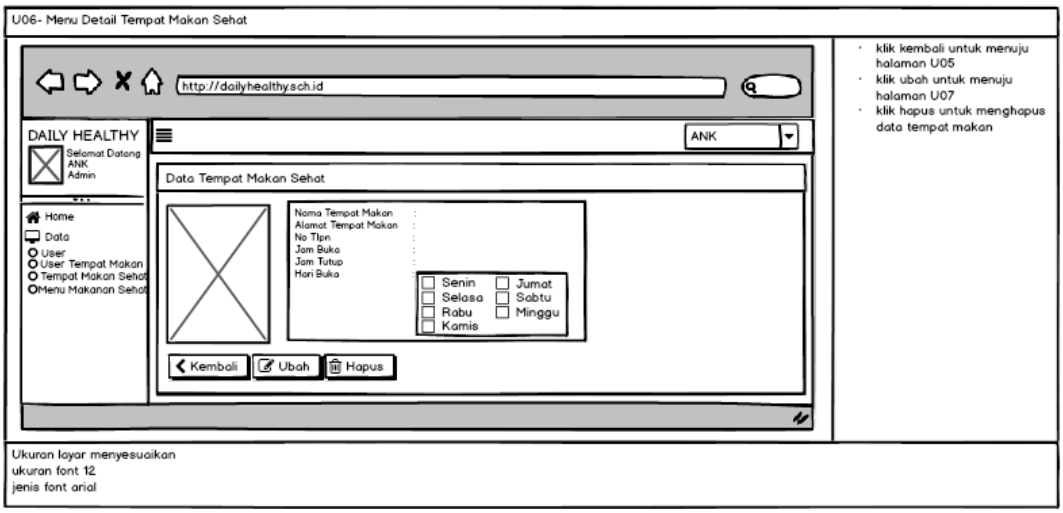
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.70 Perancangan Antarmuka Menu Utama.



Gambar 3.70 Perancangan Antarmuka Menu Utama

6. Perancangan Antarmuka Menu Detail Tempat Makan Sehat

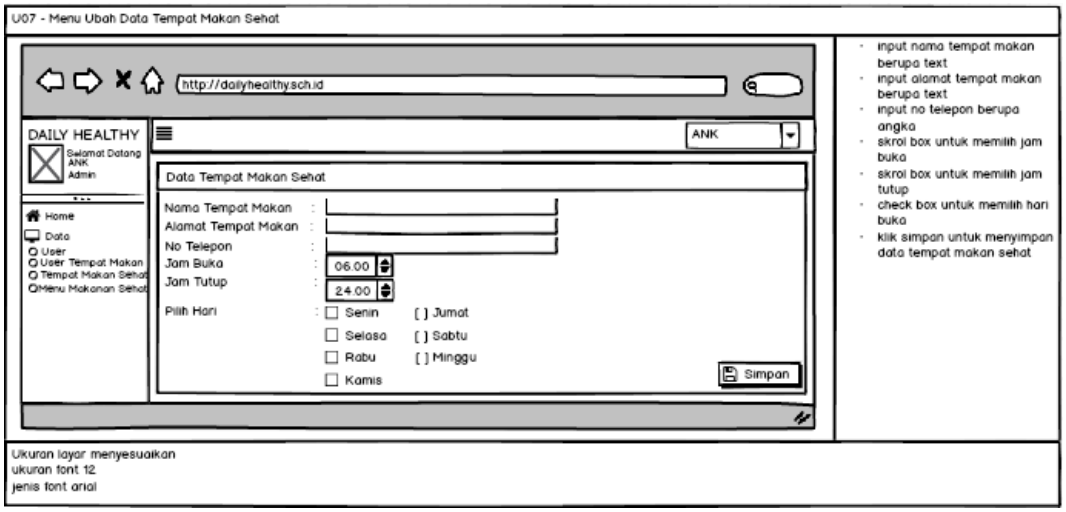
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.71 Perancangan Antarmuka Menu Detail Tempat Makan Sehat.



Gambar 3.71 Perancangan Antarmuka Menu Detail Tempat Makan Sehat

7. Perancangan Antarmuka Menu Ubah Data Tempat Makan Sehat

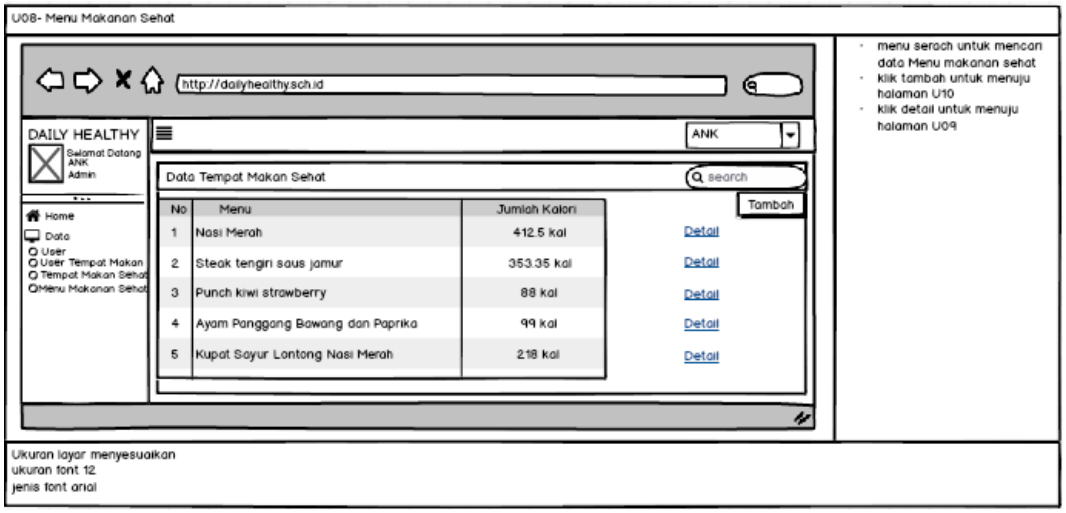
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.72 Perancangan Antarmuka Menu Ubah Data Tempat Makan Sehat.



Gambar 3.72 Perancangan Antarmuka Menu Ubah Data Tempat Makan Sehat

8. Perancangan Antarmuka Menu Data Menu Makan Sehat

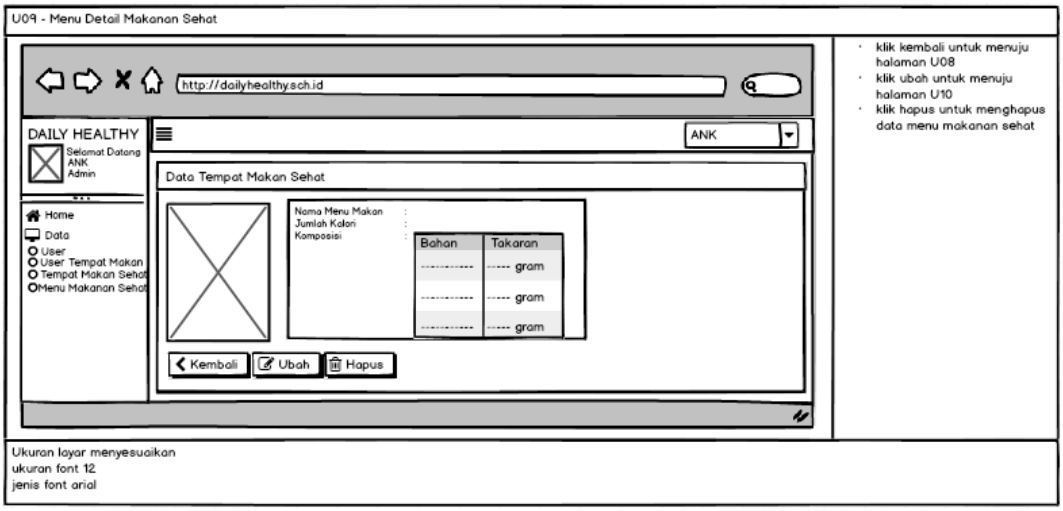
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.73 Perancangan Antarmuka Menu Data Menu Makan Sehat.



Gambar 3.73 Perancangan Antarmuka Menu Data Menu Makan Sehat

9. Perancangan Antarmuka Menu Detail Menu Makan Sehat

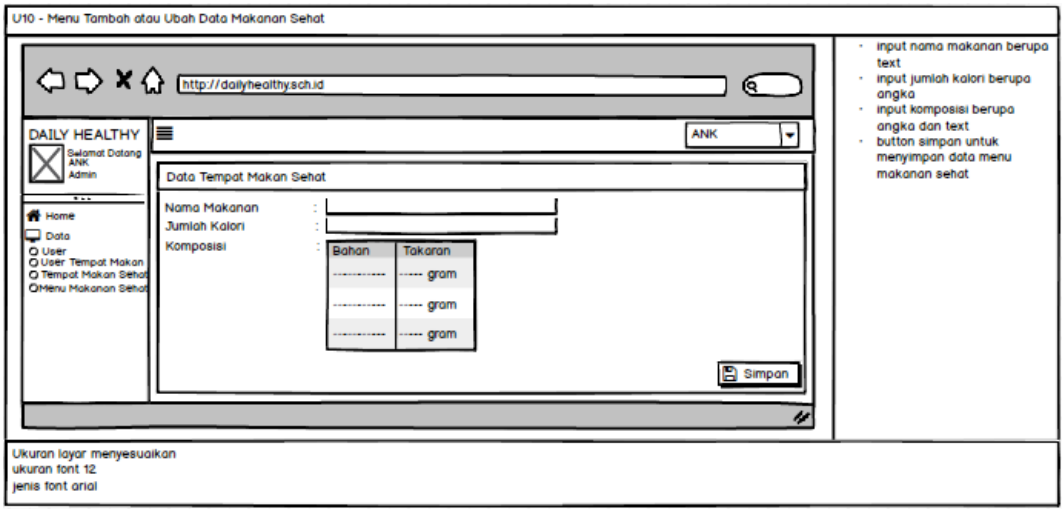
Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.74 Perancangan Antarmuka Detail Menu Makan Sehat.



Gambar 3.74 Perancangan Antarmuka Detail Menu Makan Sehat

10. Perancangan Antarmuka Menu Tambah atau Ubah Data Makan Sehat

Berikut ini merupakan antarmuka *splash screen* pada *platform web* yang dapat dilihat pada Gambar 3.75 Perancangan AntarmukaMenu Tambah atau Ubah Data Makan Sehat.



Gambar 3.75 Perancangan Antarmuka Menu Tambah atau Ubah Data Makan Sehat

3.2.4.4. Perancangan Pesan

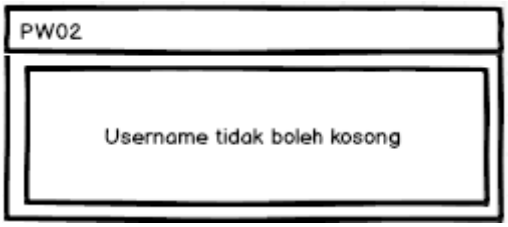
Perancangan pesan pada perangkat lunak yang dibangun terbagi menjadi dua perancangan pesan *web* dan perancangan pesan *mobile*.

1. Perancangan Pesan *Web*

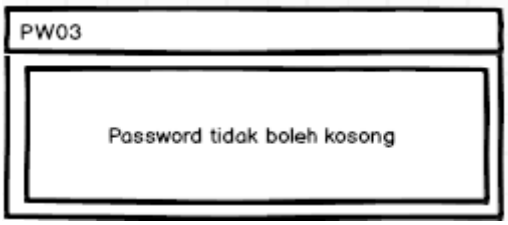
   Berikut adalah perancangan *web* pada perangkat lunak yang dibangun.



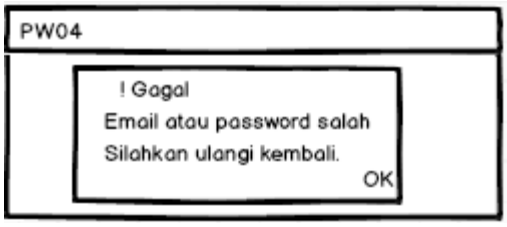
Gambar 3.76 Perancangan Pesan Web PW01



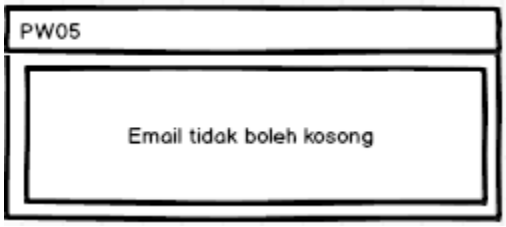
Gambar 3.77 Perancangan Pesan Web PW02



Gambar 3.78Perancangan Pesan Web PW03



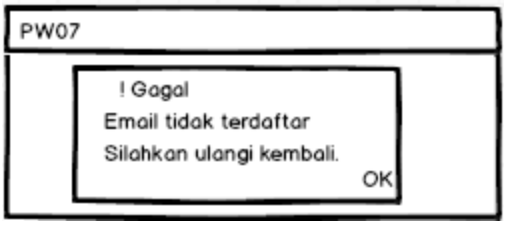
Gambar 3.79 Perancangan Pesan Web PW04



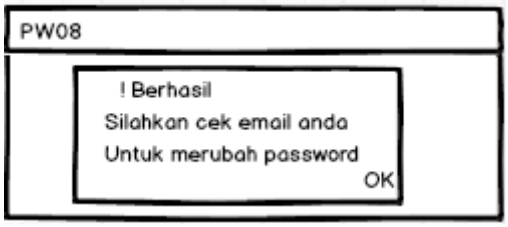
Gambar 3.80 Perancangan Pesan Web PW05



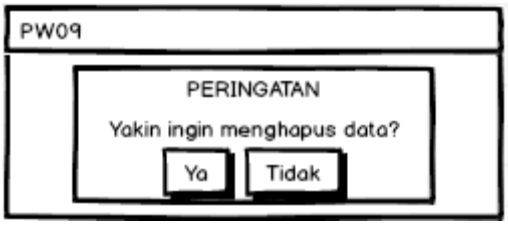
Gambar 3.81 Perancangan Pesan Web PW06



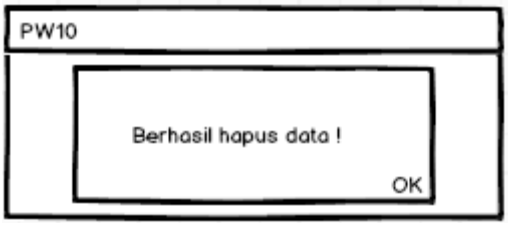
Gambar 3.82 Perancangan Pesan Web PW07



Gambar 3.83 Perancangan Pesan Web PW08



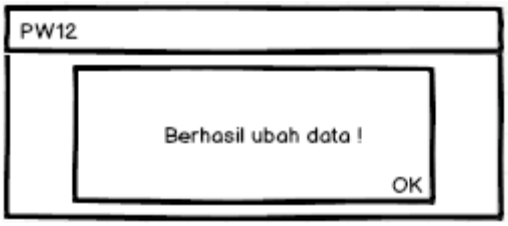
Gambar 3.84 Perancangan Pesan Web PW09



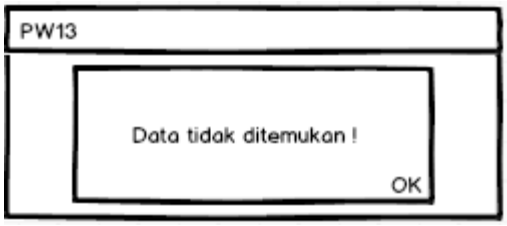
Gambar 3.85 Perancangan Pesan Web PW10



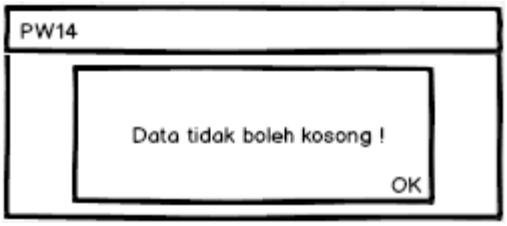
Gambar 3.86 Perancangan Pesan Web PW11



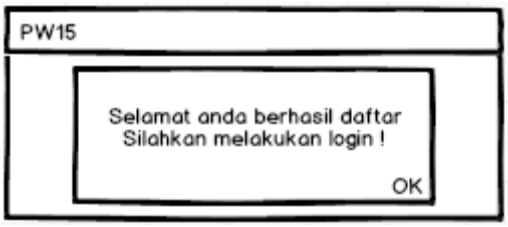
Gambar 3.87 Perancangan Pesan Web PW12



Gambar 3.88 Perancangan Pesan Web PW13



Gambar 3.89 Perancangan Pesan Web PW14



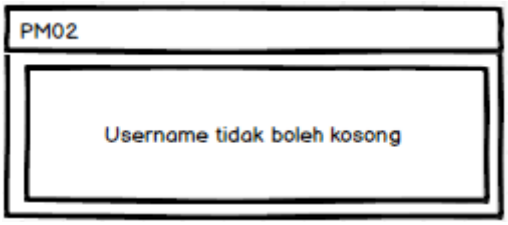
Gambar 3.90 Perancangan Pesan Web PW15

2. Perancangan Pesan *Mobile*

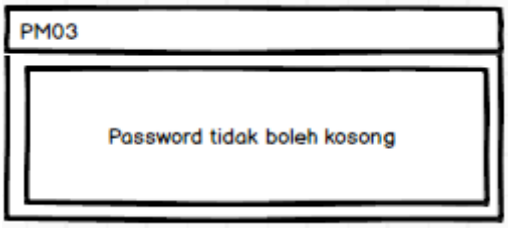
Berikut adalah perancangan *mobile* pada perangkat lunak yang dibangun.



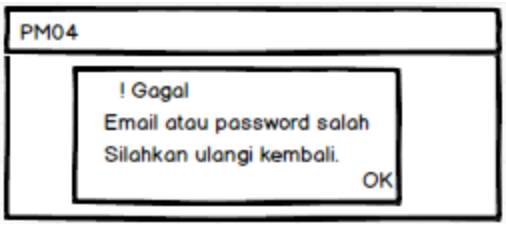
Gambar 3.91 Perancangan Pesan Mobile PM01



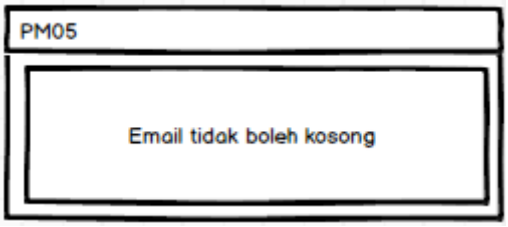
Gambar 3.92 Perancangan Pesan Mobile PM02



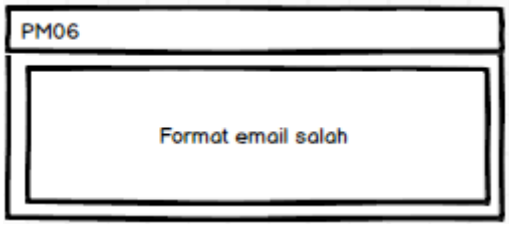
Gambar 3.93 Perancangan Pesan Mobile PM03



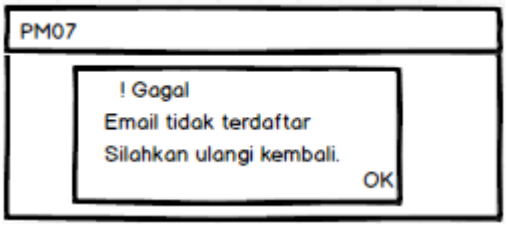
Gambar 3.94 Perancangan Pesan Mobile PM04



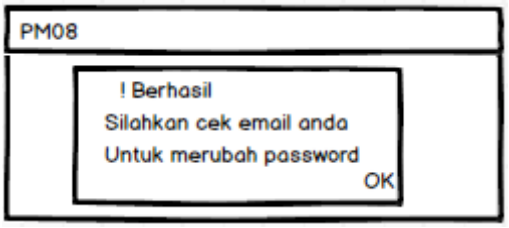
Gambar 3.95 Perancangan Pesan Mobile PM05



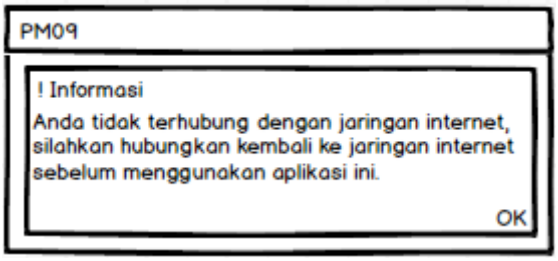
Gambar 3.96 Perancangan Pesan Mobile PM06



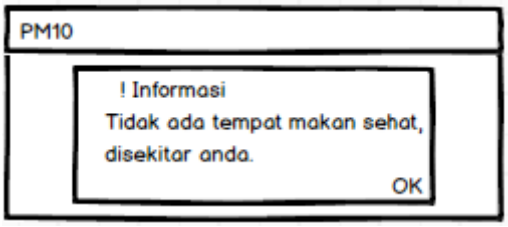
Gambar 3.97 Perancangan Pesan Mobile PM07



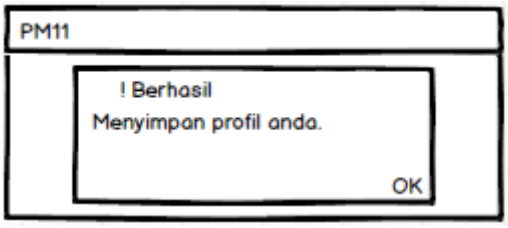
Gambar 3.98 Perancangan Pesan Mobile PM08



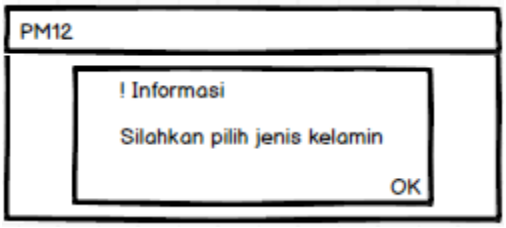
Gambar 3.99 Perancangan Pesan Mobile PM09



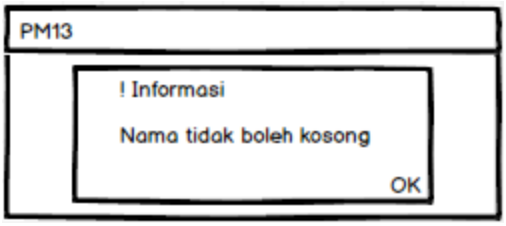
Gambar 3.100 Perancangan Pesan Mobile PM10



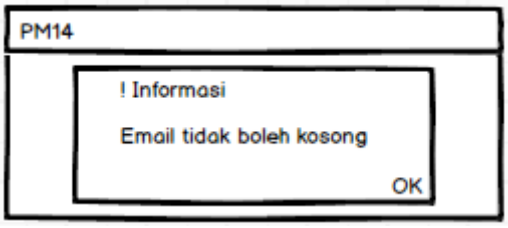
Gambar 3.101 Perancangan Pesan Mobile PM11



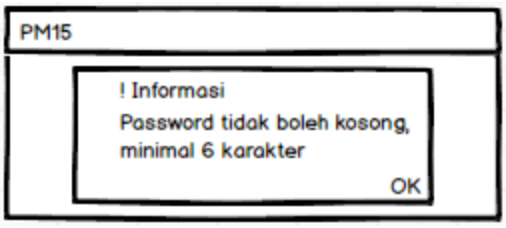
Gambar 3.102 Perancangan Pesan Mobile PM12



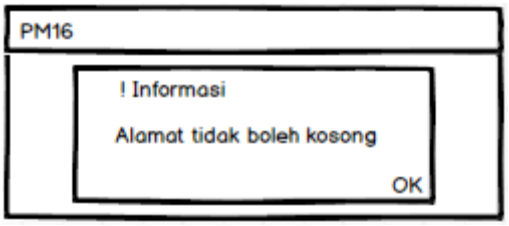
Gambar 3.103 Perancangan Pesan Mobile PM13



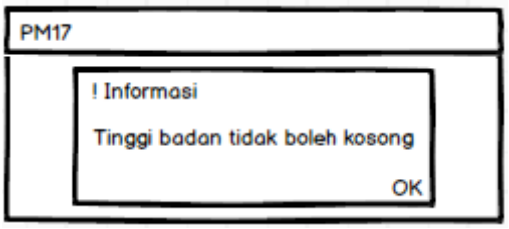
Gambar 3.104 Perancangan Pesan Mobile PM14



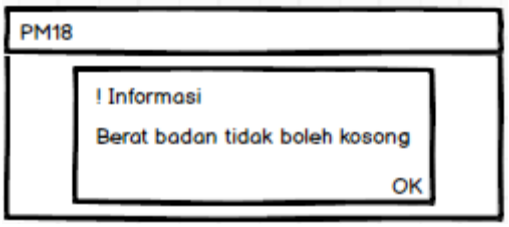
Gambar 3.105 Perancangan Pesan Mobile PM15



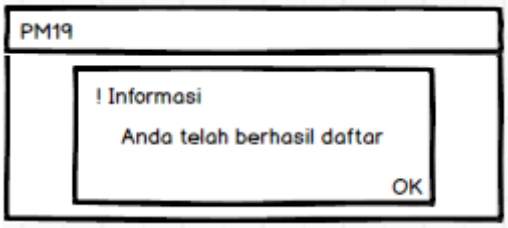
Gambar 3.106 Perancangan Pesan Mobile PM16



Gambar 3.107 Perancangan Pesan Mobile PM17



Gambar 3.108 Perancangan Pesan Mobile PM18

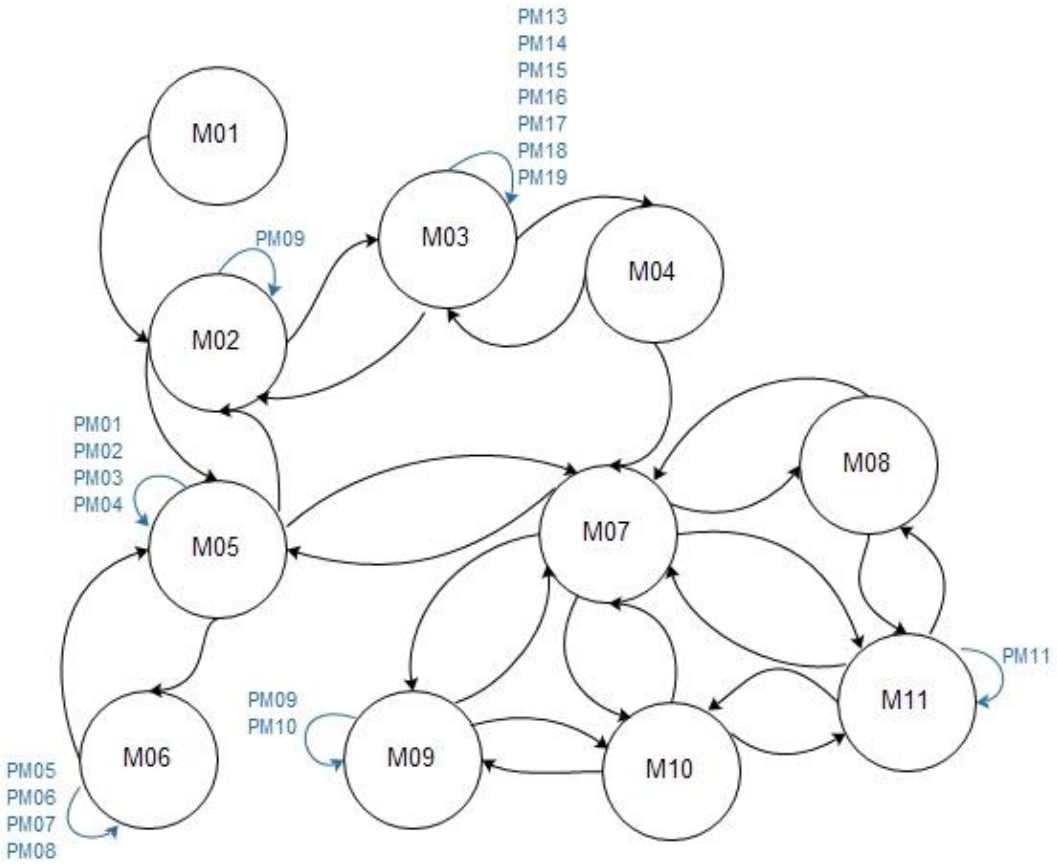


Gambar 3.109 Perancangan Pesan Mobile PM19

3.2.4.5. Jaringan Semantik

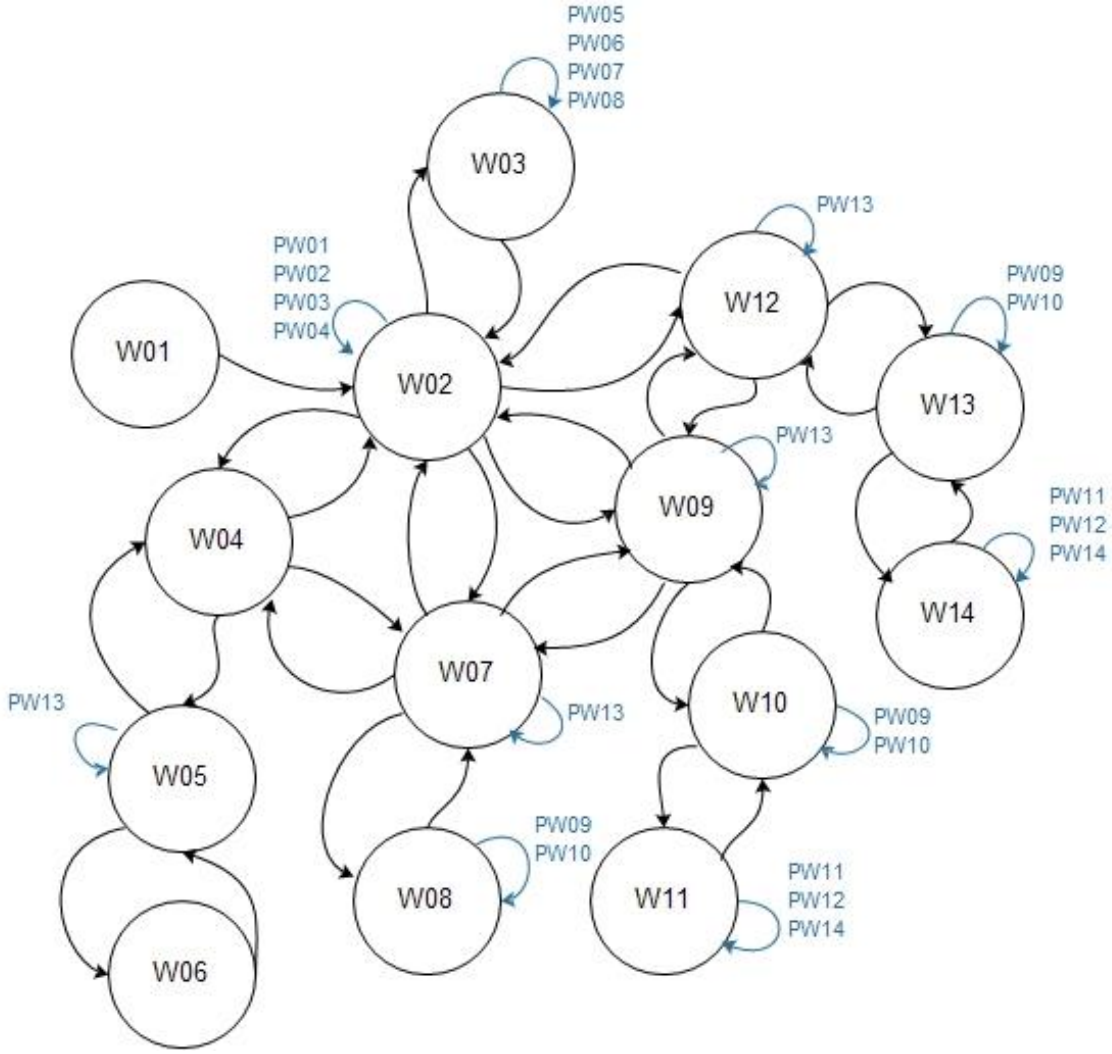
Jaringan semantik aplikasi yang akan dibangun dapat dilihat pada gambar dibawah ini :

1. Jaringan Semantik *Mobile*



Gambar 3.110 Jaringan Semantik *Mobile*

2. Jaringan Semantik *WEB Admin*



Gambar 3.111 Jaringan Semantik *WEB Admin*