

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan dijabarkan kesimpulan dan saran yang didapatkan selama proses pembangunan aplikasi tugas akhir ini.

6.1 Kesimpulan

Berdasarkan pembahasan yang telah dilakukan dari bab-bab sebelumnya, maka dapat ditarik kesimpulan yaitu aplikasi diet sehat berdasarkan golongan darah berhasil dibangun untuk membantu pengguna dalam melakukan program diet dengan memberikan informasi mengenai jumlah kebutuhan kalori harian dan kalori pada makanan berdasarkan golongan darah dengan judul PH(Pocket Healthy) terbukti dari hasil pengujian terhadap pengguna.

6.2 Saran

Berdasarkan proses pembangunan yang telah dilakukan hingga proses pengujian, penulis mendapat beberapa saran untuk pengembangan lebih terhadap aplikasi Pocket Healthy yang telah di buat sebagai berikut :

1. Pengembangan *multi platform*.
2. Sistem diharapkan bisa dikembangkan lebih dengan data yang lebih banyak dan penambahan rekomendasi menu, riwayat berat, dan fitur chatting untuk konsultasi langsung dengan pakar diet.

DAFTAR PUSTAKA

- Adinda, A. (2017). *Fakta Seputar Obesitas di Indonesia*. Retrieved from hello sehat: <https://hellosehat.com/fakta-seputar-obesitas-di-indonesia/>
- Akhmad, S. A., Resi, K. (2015). *Resep Praktis Diet Sesuai Golongan Darah A*. Yogyakarta : Citra Media Pustaka.
- Anonim, (2013). *JSON*. <http://www.json.org/json-id.html/>. diakses pada tanggal 3 Februari 2017.
- _____, (2014). *Tingkat obesitas Indonesia nomor 10 dunia*. Retrieved from Situs Web BBC Indonesia: http://www.bbc.com/indonesia/majalah/2014/05/140529_iptek_indonesia_obesitas
- Gusti, F., dan Adil, R. (2010). Pembuatan Prototype Alat Identifikasi Golongan Darah pada Manusia Berbasis Pola Sidik Jari Menggunakan Scanner Optik. *Politeknik Elektronika Negeri Surabaya*, 1-8.
- Halls S.B. (2005) : About arithmetic formulas for calculating ideal body weight . <http://halls.md/ideal-weight-formulas-broca-devine/>. Diakses pada tanggal 10 Maret 2017.
- Hikmah, M., Sarwosri, S., & Sunaryono, D. (2013). Aplikasi Rekomendasi Menu Makan untuk Penderita Kanker Berdasarkan Golongan Darah Berbasis Android pada Perangkat Komunikasi Bergerak dengan Menggunakan Metode Forward Chaining. *Jurnal Teknik Pomits, II(1)*, 1 - 4.
- Kadir, A. (2013). *From Zero to a Pro Pemrograman Aplikasi Android*. Yogyakarta: Andi Yogyakarta.
- Nesia, H. (2010). *Bangunan Aplikasi diet golongan darah berbasis multi media*. Yogyakarta: Universitas AtmaJaya.
- Novita, I. E., & Sulistyanto, H. (2015). *Pengembangan Aplikasi Untuk Mengetahui Kebutuhan Jumlah Kalori*. Skripsi, Universitas Muhammadiyah Surakarta.
- Pamudji, D. (2011). *Aplikasi Penghitung Kalori Makanan Berbasis Android 2.1*. Jurnal Skripsi.
- Putra, R. B., Sutardji, S., & Woro, O. (2011). Sistem Informasi Perencanaan Pola Hidup Sehat melalui Keseimbangan Aktivitas dan Asupan Makanan. *Jurnal Media Ilmu Keolahragaan Indonesia, I(2)*, 181 - 194.

Rahman, M. A., Kuswardayan, I. & Hariadi, R. R., 2013. Perancangan dan Implementasi RESTful Web Service untuk Game Sosial Food Merchant Saga pada Perangkat Android. *Jurnal Teknik Pomits*, II(1), pp. 1-4.

Sanjaya, I. (2014). Pengembangan Aplikasi Rekomendasi Diet Berbasis Android. Yogyakarta: UniverAtmaJaya.

Savitri, A. (2016). Diet Sehat Mudah Berdasarkan Golongan Darah. Yogyakarta : Pustaka Baru Press.



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Pocket Healthy

(PH)

Untuk :

**Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

Dipersiapkan Oleh :

Fery Wijaya / 120706802

**Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<i>DPPL - PH</i>		1/86
		Revisi	A	

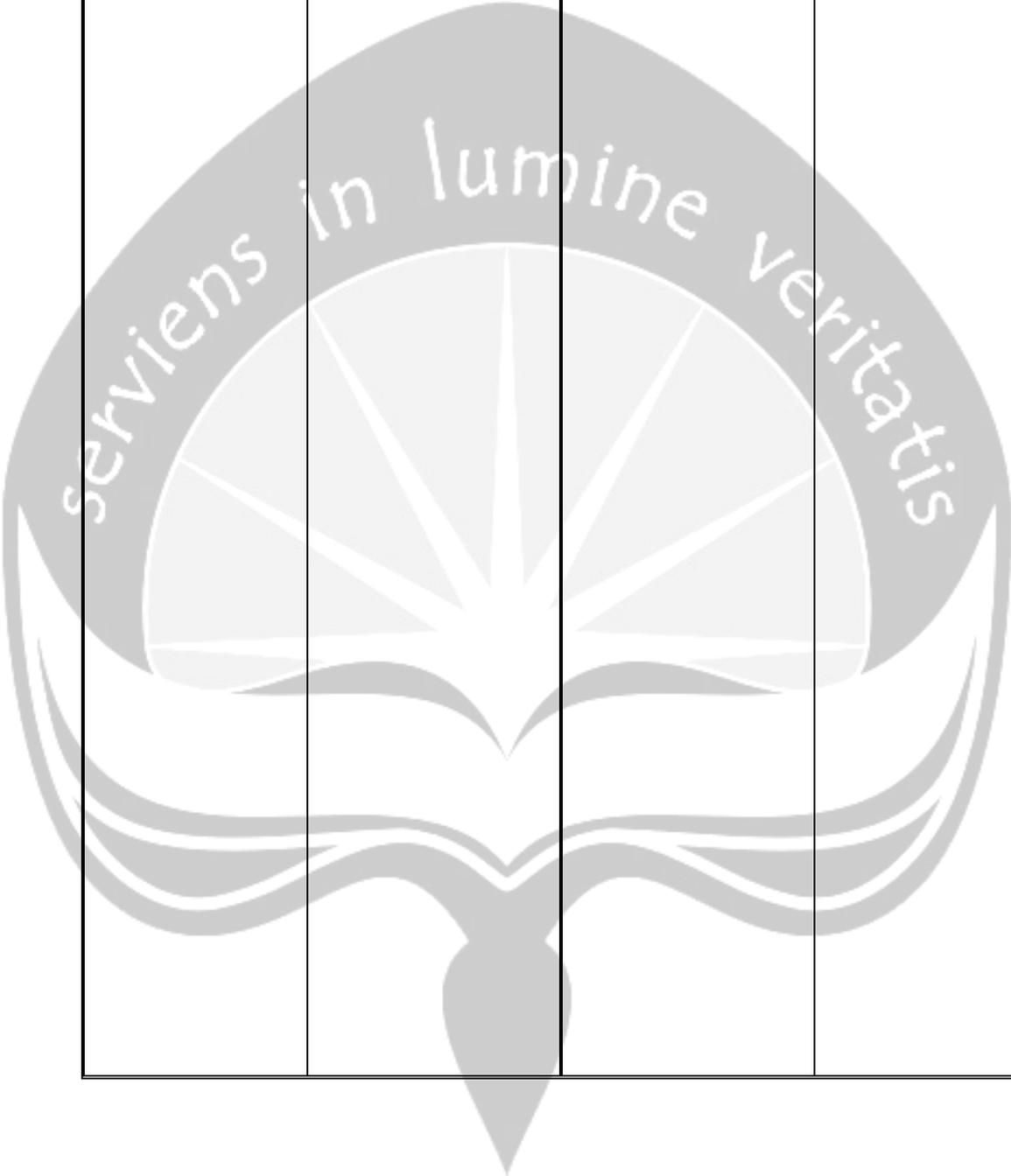
DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

INDEKS TGL	-	A	B	C	D	E	F
DITULIS OLEH	FW						
DIPERIKSA OLEH	YSP ERN						
DISETUJUI OLEH	YSP ERN						

DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi



DAFTAR ISI

1. Pendahuluan	8
1.1 Tujuan	8
1.2 Lingkup Masalah	8
1.3 Definisi, Akronim dan Singkatan	8
1.4 Referensi	10
2. Perancangan Sistem	12
2.1 Perancangan Asitektur	12
2.1.1 Web Pocket Healthy	12
2.1.2 Mobile Pocket Healthy	13
2.2 Perancangan Rinci	14
2.2.1 Sequence Diagram Web Pocket Healthy	14
2.2.1.1 Fungsi Login	14
2.2.1.2 Fungsi Mengedit Profil	14
2.2.1.3 Fungsi Tambah Data Kelompok Makanan	15
2.2.1.4 Fungsi Edit Data Kelompok Makanan	15
2.2.1.5 Fungsi Tampil Data Kelompok Makanan	16
2.2.1.6 Fungsi Hapus Data Kelompok Makanan	16
2.2.1.7 Fungsi Tambah Data Daftar Makanan	17
2.2.1.8 Fungsi Edit Data Daftar Makanan	17
2.2.1.9 Fungsi Tampil Data Daftar Makanan	18
2.2.1.10 Fungsi Hapus Data Daftar Makanan	18
2.2.1.11 Fungsi Tambah Data Jenis Makanan	19
2.2.1.12 Fungsi Edit Data Jenis Makanan	19
2.2.1.13 Fungsi Tampil Data Jenis Makanan	20
2.2.1.14 Fungsi Hapus Data Jenis Makanan	20
2.2.1.15 Fungsi Tampil Data Pengguna	21
2.2.1.16 Fungsi Hapus Data Pengguna	21
2.2.2 Sequence Diagram Mobile Pocket Healthy	22
2.2.2.1 Fungsi Login Pengguna	22
2.2.2.2 Fungsi Daftar Pengguna	23
2.2.2.3 Fungsi Tampil Profil	23
2.2.2.4 Fungsi Jadwal Makan	24
2.2.2.4 Fungsi Tampil Kelompok Makanan	24
2.2.2.5 Fungsi Tampil Daftar Makanan	25
2.2.2.6 Fungsi Tampil Detail Makanan	25
2.2.2.7 Fungsi Hitung Berat Ideal	26
2.2.2.8 Fungsi Hitung Kebutuhan Kalori	27
2.2.3 Class Diagram	28
2.2.3.1 Class Diagram Web PH	28
2.2.3.2 Class Diagram Pocket Healthy	30
2.2.4 Class Diagram Specific Descriptions Web PH	31
2.2.4.2 Specific Design Class login	31
2.2.4.4 Specific Design Class dashboard	31

2.2.4.5	Specific Design Class kelompok_makanan	31
2.2.4.6	Specific Design Class tambah_kelompok_makanan	32
2.2.4.7	Specific Design Class ubah_kelompok_makanan	32
2.2.4.8	Specific Design Class hapus_kelompok_makanan	32
2.2.4.9	Specific Design Class daftar_makanan	33
2.2.4.10	Specific Design Class tambah_daftar_makanan	33
2.2.4.11	Specific Design Class ubah_daftar_makanan	33
2.2.4.12	Specific Design Class hapus_daftar_makanan	33
2.2.4.13	Specific Design Class jenis_makanan	34
2.2.4.14	Specific Design Class tambah_jenis_makanan	34
2.2.4.15	Specific Design Class ubah_jenis_makanan	34
2.2.4.16	Specific Design Class hapus_jenis_makanan	35
2.2.4.17	Specific Design Class pengguna	35
2.2.4.18	Specific Design Class Auth	35
2.2.4.19	Specific Design Class Profil	36
2.2.4.20	Specific Design Class KelompokMakanan	36
2.2.4.21	Specific Design Class DaftarMakanan	37
2.2.4.21	Specific Design Class JenisMakanan	37
2.2.4.19	Specific Design Class pengguna_control	38
2.2.4.20	Specific Design Class admin	38
2.2.4.22	Specific Design Class kelompok_makanan	39
2.2.4.23	Specific Design Class daftar_makanan	40
2.2.4.24	Specific Design Class jenis_makanan	41
2.2.4.12	Specific Design Class pengguna	42
2.2.5	Class Diagram Specific Descriptions Mobile PH	44
2.2.5.1	Specific Design Class Login UI	44
2.2.5.2	Specific Design Class Daftar UI	44
2.2.5.3	Specific Design Class Profil UI	44
2.2.5.4	Specific Design Class Jadwal Makan UI	45
2.2.5.5	Specific Design Class Kelompok Makanan UI	45
2.2.5.6	Specific Design Class Daftar Makanan UI	45
2.2.5.7	Specific Design Class Detail Makanan UI	46
2.2.5.8	Specific Design Class Hitung Berat Ideal UI	46
2.2.5.8	Specific Design Class Hitung Kebutuhan Kalori UI	46
2.2.5.9	Specific Design Class Login Activity	47
2.2.5.10	Specific Design Class Daftar Activity	47
2.2.5.11	Specific Design Class Profil Activity	47
2.2.5.12	Specific Design Class Jadwal Makan Activity	48
2.2.5.13	Specific Design Class Kelompok Makanan Activity	48
2.2.5.14	Specific Design Class Daftar Makanan Activity	48
2.2.5.15	Specific Design Class Detail Makanan Activity	49
2.2.5.16	Specific Design Class Hitung Berat Ideal Activity	49
2.2.5.17	Specific Design Class Hitung Kebutuhan Kalori Harian Activity	49

2.2.5.18 Specific Design Class PHJsonApi	50
3. Perancangan Data	52
3.1 Dekomposisi Data	52
4.1.1. Sketsa UI Perangkat Lunak Web PH dan Deskripsinya	57
4.1.1.1. Antarmuka Login	57
4.1.1.4. Antarmuka Kelola Kelompok Makanan	60
4.1.1.5. Antarmuka Tambah Kelompok Makanan	61
4.1.1.6. Antarmuka Ubah Kelompok Makanan	62
4.1.1.7. Antarmuka Kelola Daftar Makanan	63
4.1.1.8. Antarmuka Tambah Daftar Makanan	64
4.1.1.9. Antarmuka Ubah Daftar Makanan	65
4.1.1.10. Antarmuka Kelola Jenis Makanan	66
4.1.1.11. Antarmuka Tambah Jenis Makanan	67
4.1.1.12. Antarmuka Ubah Jenis Makanan	68
4.1.1.13. Antarmuka Kelola Pengguna	69
4.1.2. Sketsa UI Perangkat Lunak Mobile PH dan Deskripsinya	70
4.1.2.1. Antarmuka Halaman Splash	70
4.1.2.2. Antarmuka Halaman Intro	71
4.1.2.3. Antarmuka Halaman Login	72
4.1.2.4. Antarmuka Halaman Mendaftar	73
4.1.2.5. Antarmuka Halaman Beranda	75
4.1.2.6. Antarmuka Update Informasi Pribadi	76
4.1.2.7. Antarmuka Navigation Drawer / Menu	77
4.1.2.8. Antarmuka Halaman Jadwal Makan	78
4.1.2.9. Antarmuka Halaman Kelompok Makanan	79
4.1.2.10. Antarmuka Halaman Daftar Makanan	80
4.1.2.11. Antarmuka Halaman Jenis Makanan	81
4.1.2.12. Antarmuka Hitung Kebutuhan Kalori Harian	82
4.1.2.13. Antarmuka Hitung Indeks Massa Tubuh	83
4.1.2.14. Antarmuka Tentang Aplikasi	84
4.1.2.15. Antarmuka Pengaturan Akun	85
4.1.2.16. Antarmuka Detail Jenis Makanan	86

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur Web Pocket Healthy	12
Gambar 2. 2 Arsitektur Mobile Pocket Healthy	13
Gambar 2. 3 Sequence Diagram Login	14
Gambar 2. 4 Sequence Diagram Mengedit Profil	14
Gambar 2. 5 Sequence Diagram Tambah Data Kelompok Makanan	15
Gambar 2. 6 Sequence Diagram Edit Data Kelompok Makanan ..	15
Gambar 2. 7 Sequence Diagram Tampil Data Kelompok Makanan	16
Gambar 2. 8 Sequence Diagram Hapus Data Kelompok Makanan .	16
Gambar 2. 9 Sequence Diagram Tambah Data Daftar Makanan ..	17
Gambar 2. 10 Sequence Diagram Edit Data Daftar Makanan ...	17

Program Studi Teknik Informatika	DPPL – PH	6/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Gambar 2. 11	Sequence Diagram Tampil Data Daftar Makanan .	18
Gambar 2. 12	Sequence Diagram Hapus Data Daftar Makanan ..	18
Gambar 2. 13	Sequence Diagram Tambah Data Jenis Makanan ..	19
Gambar 2. 14	Sequence Diagram Edit Data Jenis Makanan	19
Gambar 2. 15	Sequence Diagram Tampil Data Jenis Makanan ..	20
Gambar 2. 16	Sequence Diagram Hapus Data Jenis Makanan ...	20
Gambar 2. 17	Sequence Diagram Tampil Data Pengguna	21
Gambar 2. 18	Sequence Diagram Hapus Data Pengguna	21
Gambar 4. 1	Antarmuka Login	57
Gambar 4. 2	Antarmuka Beranda	58
Gambar 4. 3	Antarmuka Edit Profil	59
Gambar 4. 4	Antarmuka Kelola Kelompok Makanan	60
Gambar 4. 5	Antarmuka Tambah Kelompok Makanan	61
Gambar 4. 6	Antarmuka Ubah Kelompok Makanan	62
Gambar 4. 7	Antarmuka Kelola Daftar Makanan	63
Gambar 4. 12	Antarmuka Tambah Daftar Makanan	64
Gambar 4. 9	Antarmuka Ubah Daftar Makanan	65
Gambar 4. 10	Antarmuka Kelola Jenis Makanan	66
Gambar 4. 11	Antarmuka Tambah Jenis Makanan	67
Gambar 4. 12	Antarmuka Ubah Jenis Makanan	68
Gambar 4. 13	Antarmuka Kelola Pengguna	69
Gambar 4. 14	Antarmuka Halaman Splash	70
Gambar 4. 15	Antarmuka Halaman Intro	71
Gambar 4. 16	Antarmuka Halaman Login	72
Gambar 4. 17	Antarmuka Halaman Mendaftar	73
Gambar 4. 18	Antarmuka Halaman Mendaftar	74
Gambar 4. 19	Antarmuka Halaman Beranda	75
Gambar 4. 20	Antarmuka Update Informasi Pribadi	76
Gambar 4. 21	Antarmuka Navigation Drawer	77
Gambar 4. 22	Antarmuka Jadwal Makan	78
Gambar 4. 23	Antarmuka Kelompok Makanan	79
Gambar 4. 24	Antarmuka Halaman Daftar Makanan	80
Gambar 4. 25	Antarmuka Halaman Jenis Makanan	81
Gambar 4. 26	Antarmuka Hitung Kebutuhan Kalori Harian	82
Gambar 4. 27	Antarmuka Hitung Indeks Massa Tubuh	83
Gambar 4. 28	Antarmuka Tentang Aplikasi	84
Gambar 4. 29	Antarmuka Pengaturan Akun	85
Gambar 4. 30	Antarmuka Dialog Tambah Makanan	86

1. Pendahuluan

1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini bertujuan untuk mendefinisikan perancangan perangkat lunak Pocket Healthy (PH) yang akan dikembangkan. Dokumen tersebut akan digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap berikutnya.

1.2 Lingkup Masalah

Perangkat Lunak PH dikembangkan dengan tujuan untuk:

1. Mengelola data makanan berdasarkan golongan darah dan melihat data pengguna berbasis web.
2. Menghitung dan menampilkan hasil dari indeks massa tubuh dan kebutuhan kalori harian berdasarkan masukan data berat, tinggi, umur, dan jenis aktivitas dari pengguna berbasis *mobile*.
3. Membantu mengatur jadwal makan harian dengan menu makanan berdasarkan golongan darah serta membatasi maksimal kalori harian sesuai data pengguna berbasis *mobile*.
4. Memudahkan pengguna dalam menjalankan diet sehat berdasarkan golongan darah.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD).

Program Studi Teknik Informatika	DPPL – PH	8/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

	Merupakan deskripsi dari perancangan produk /perangkat lunak yang akan dikembangkan.
PH	Perangkat lunak untuk memudahkan diet sehat berdasarkan golongan darah.
DPPL-PH-XXX	Kode yang merepresentasikan kebutuhan pada PH(Aplikasi mobile Pocket Healthy) dimana XXX merupakan nomor fungsi produk.
DPPL-PH-WEB-XXX	Kode yang merepresentasikan kebutuhan pada PH-WEB (Web untuk pengelolaan data untuk aplikasi mobile)dimana XXX merupakan nomor fungsi produk.
Web	Sejumlah halaman yang memiliki topik yang terkait, dapat juga disertai dengan gambar, video atau jenis lainnya. Biasanya dapat diakses menggunakan jaringan internet ataupun jaringan wilayah lokal (LAN)
Android	Sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layarsentuhseperti telepon pintar dan komputer tablet.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global

	yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
PHP	Hypertext Preprocessor merupakan Bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Java	JAVA merupakan sebuah Bahasa pemrograman yang berorientasi OBJEK dan dapat dijalankan (<i>Run</i>) pada segala jenis sistem operasi (OS) karena perkembangan JAVA ini tidak hanya terfokus pada satu sistem operasi saja tetapi dikembangkan untuk segala macam jenis sistem operasi dan bersifat <i>open source</i> .
XML	Bahasa markup untuk keperluan umum yang disarankan oleh W3C untuk membuat dokumen markup keperluan pertukaran data antar sistem yang beraneka ragam.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

Program Studi Teknik Informatika	DPPL – PH	10/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

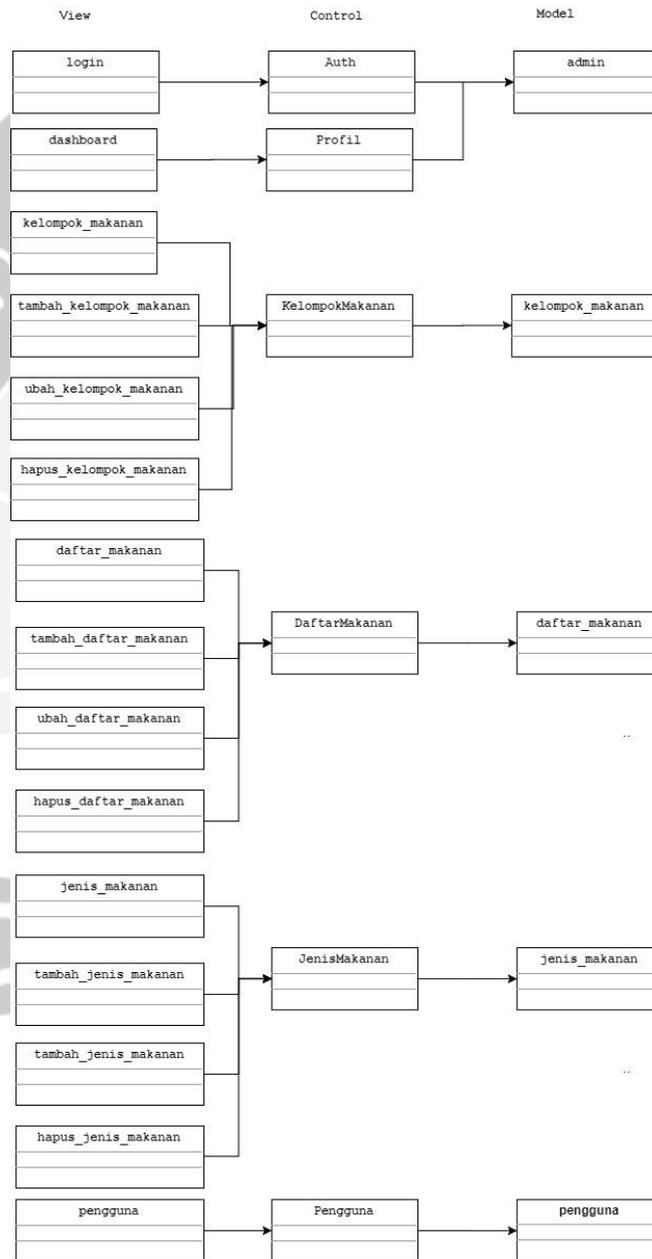
1. Fery Wijaya / 6802, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) PH*, Universitas Atma Jaya Yogyakarta. Perancangan Sistem



2. Perancangan Sistem

2.1 Perancangan Asitektur

2.1.1 Web Pocket Healthy

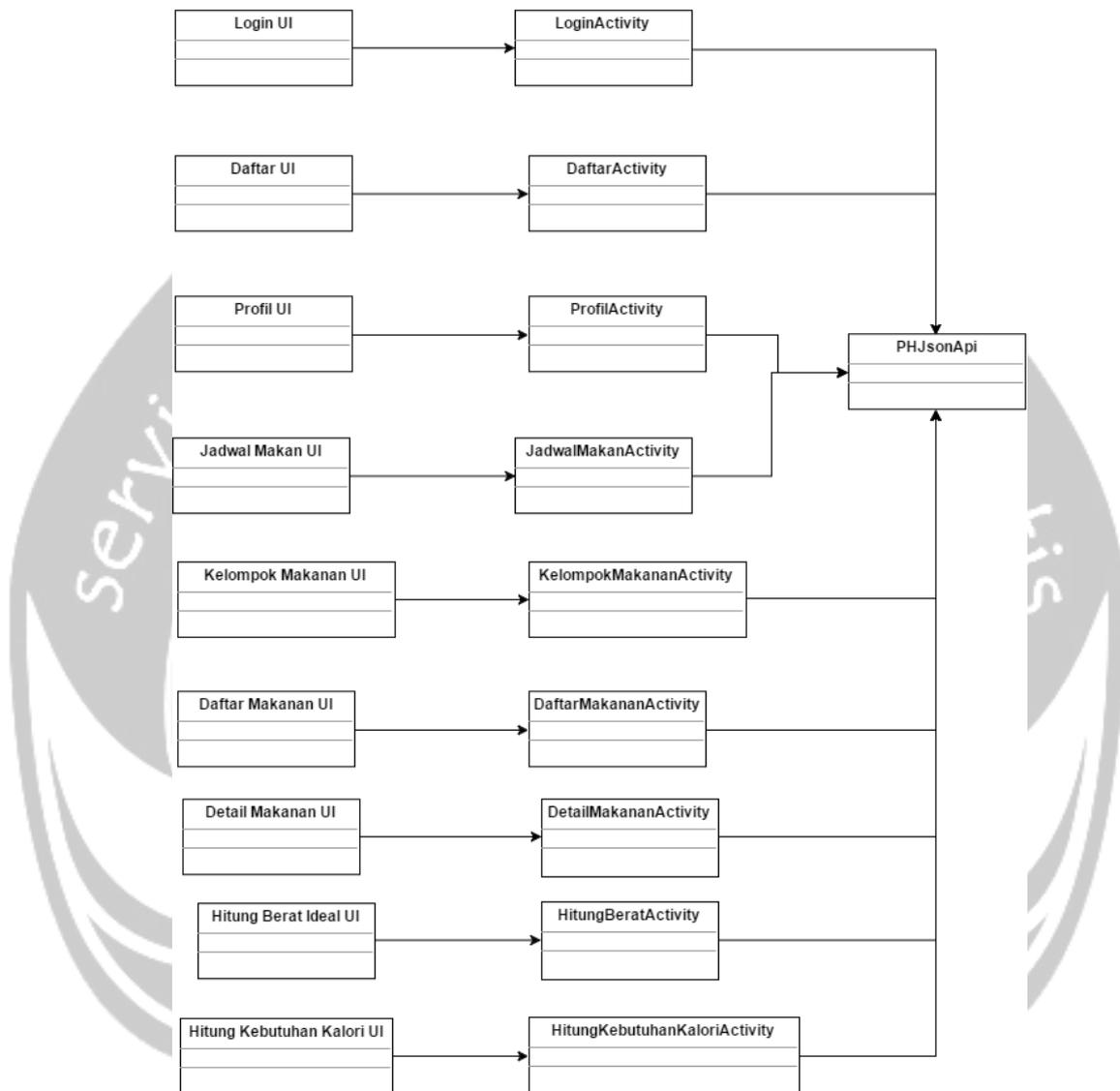


Gambar 2. 1 Arsitektur Web Pocket Healthy

Gambar 2.1 merupakan perancangan arsitektur pada web Pocket Healthy.

Program Studi Teknik Informatika	DPPL – PH	12/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2.1.2 Mobile Pocket Healthy



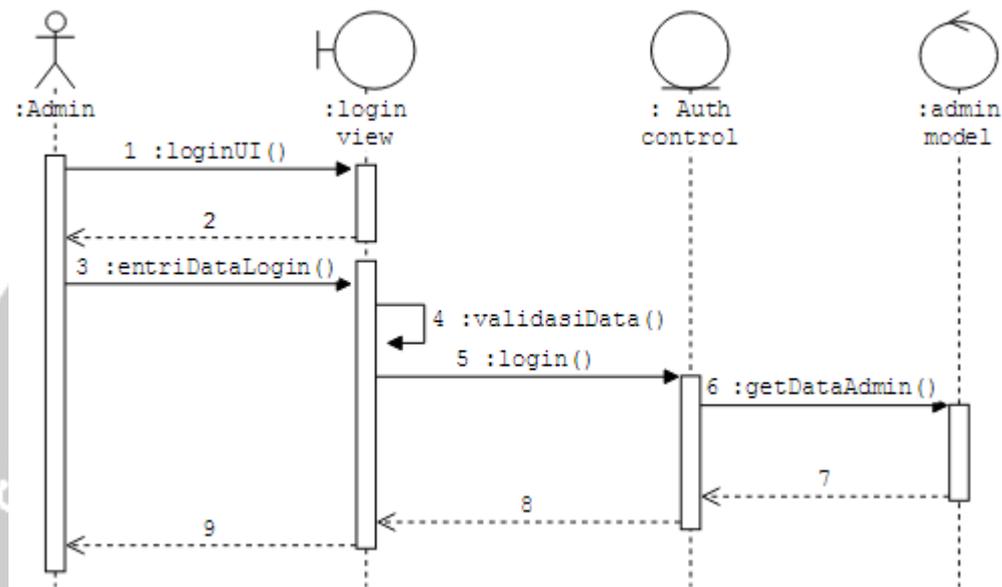
Gambar 2. 2 Arsitektur Mobile Pocket Healthy

Gambar 2.2 merupakan perancangan arsitektur pada mobile Pocket Healthy.

2.2 Perancangan Rinci

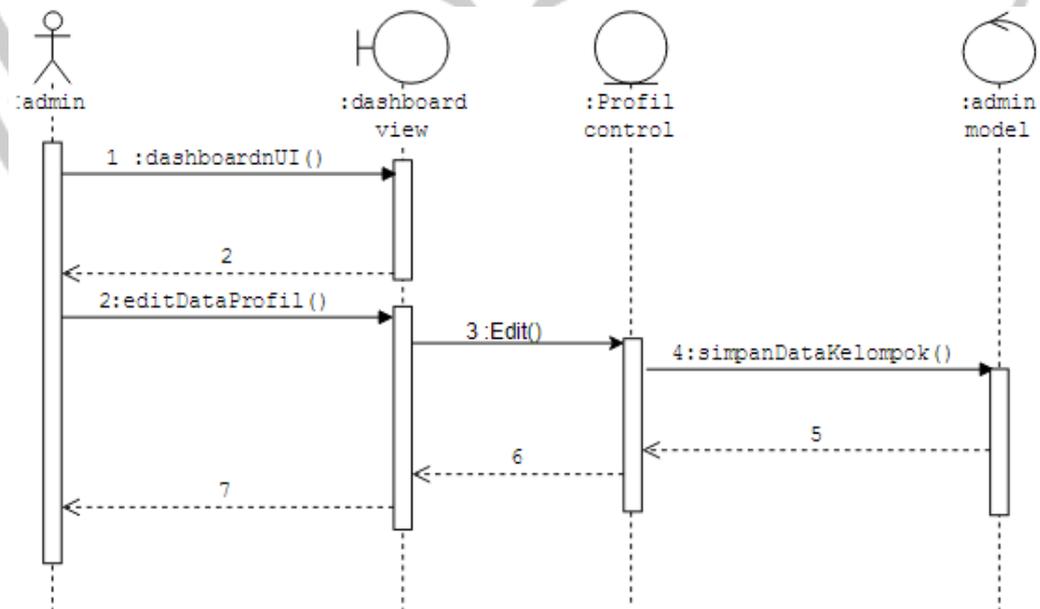
2.2.1 Sequence Diagram Web Pocket Healthy

2.2.1.1 Fungsi Login



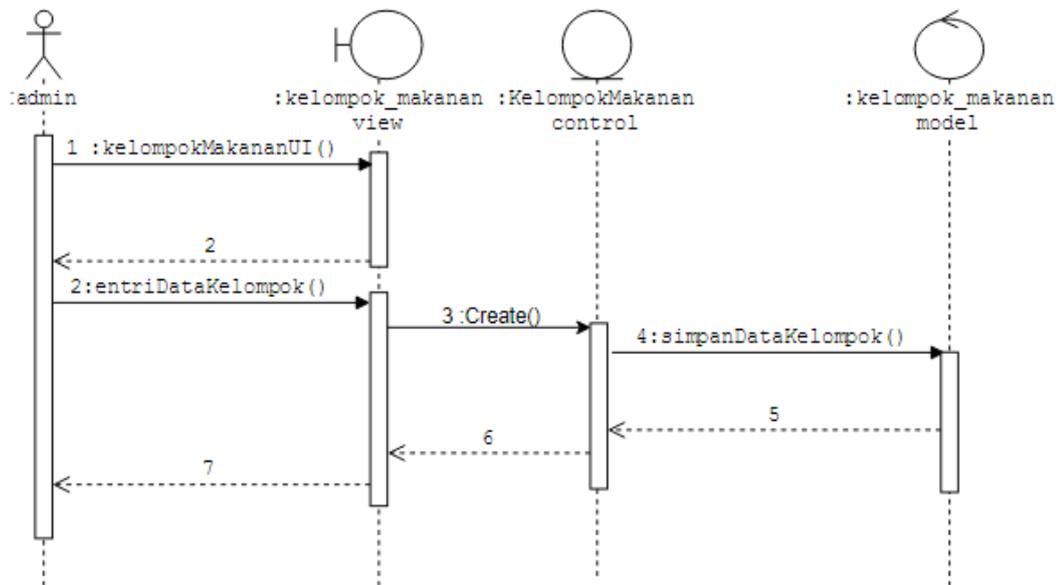
Gambar 2. 3 Sequence Diagram Login

2.2.1.2 Fungsi Mengedit Profil



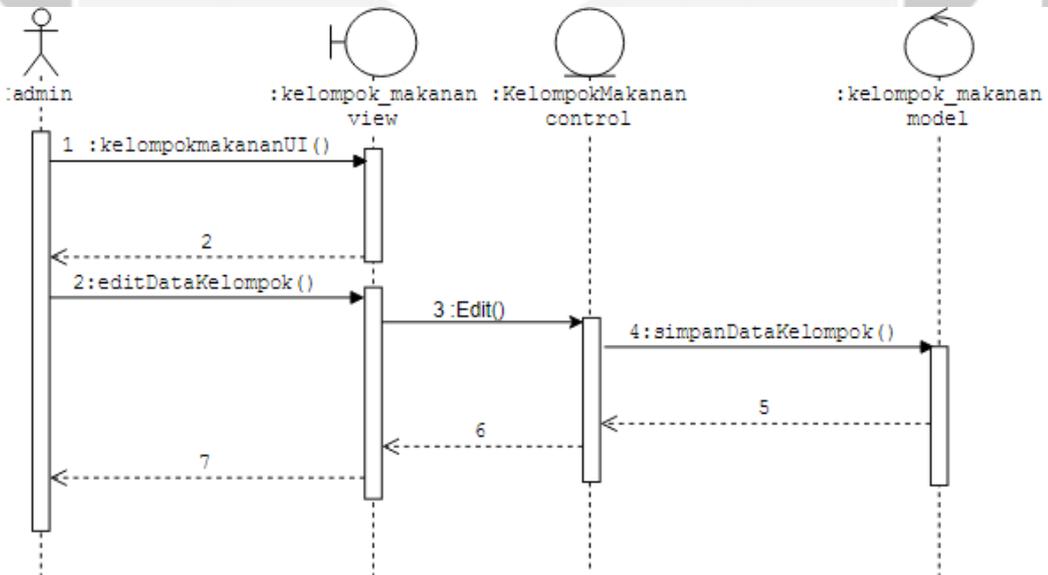
Gambar 2. 4 Sequence Diagram Mengedit Profil

2.2.1.3 Fungsi Tambah Data Kelompok Makanan



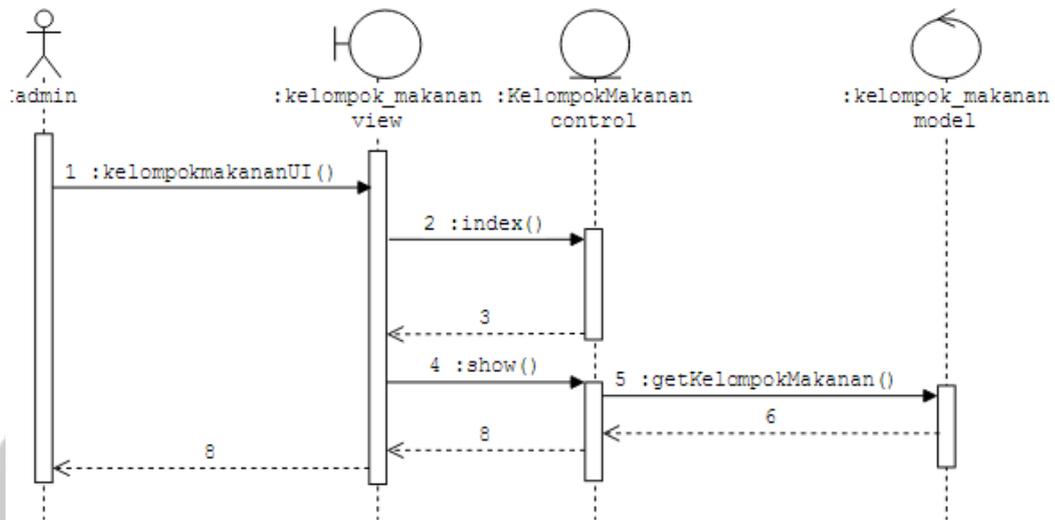
Gambar 2. 5 Sequence Diagram Tambah Data Kelompok Makanan

2.2.1.4 Fungsi Edit Data Kelompok Makanan



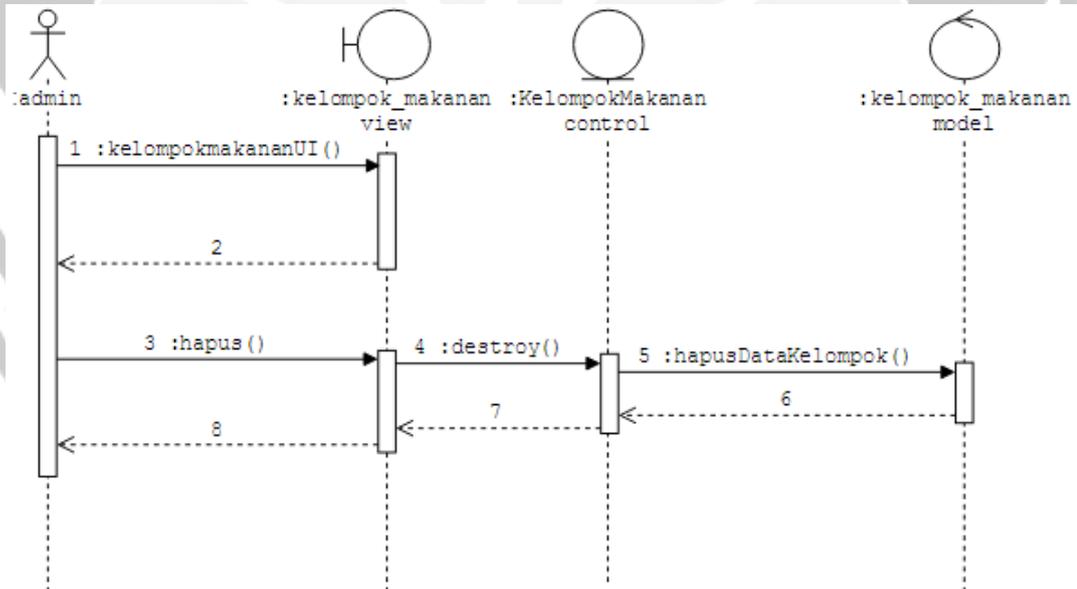
Gambar 2. 6 Sequence Diagram Edit Data Kelompok Makanan

2.2.1.5 Fungsi Tampil Data Kelompok Makanan



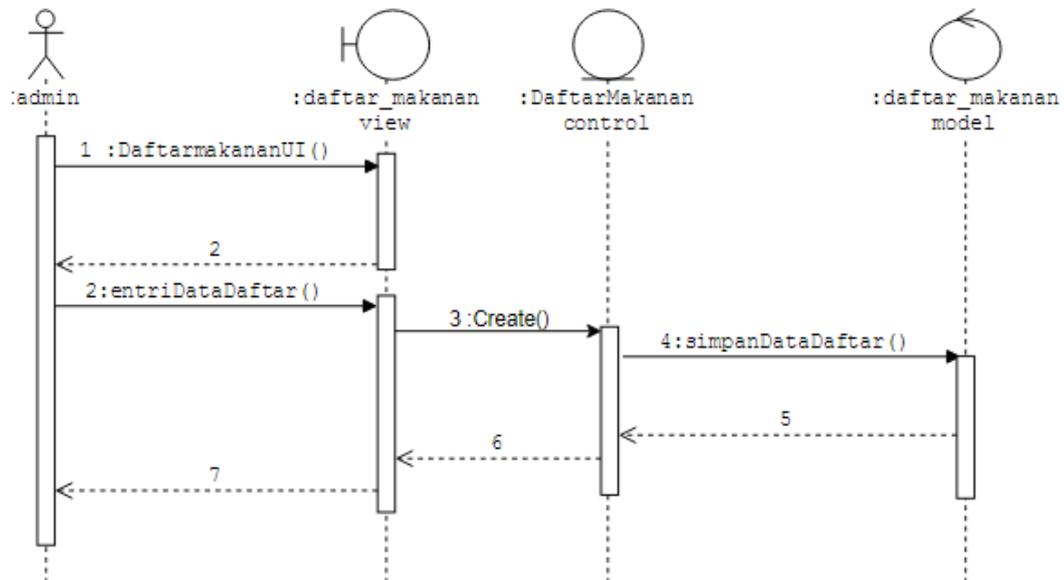
Gambar 2. 7 Sequence Diagram Tampil Data Kelompok Makanan

2.2.1.6 Fungsi Hapus Data Kelompok Makanan



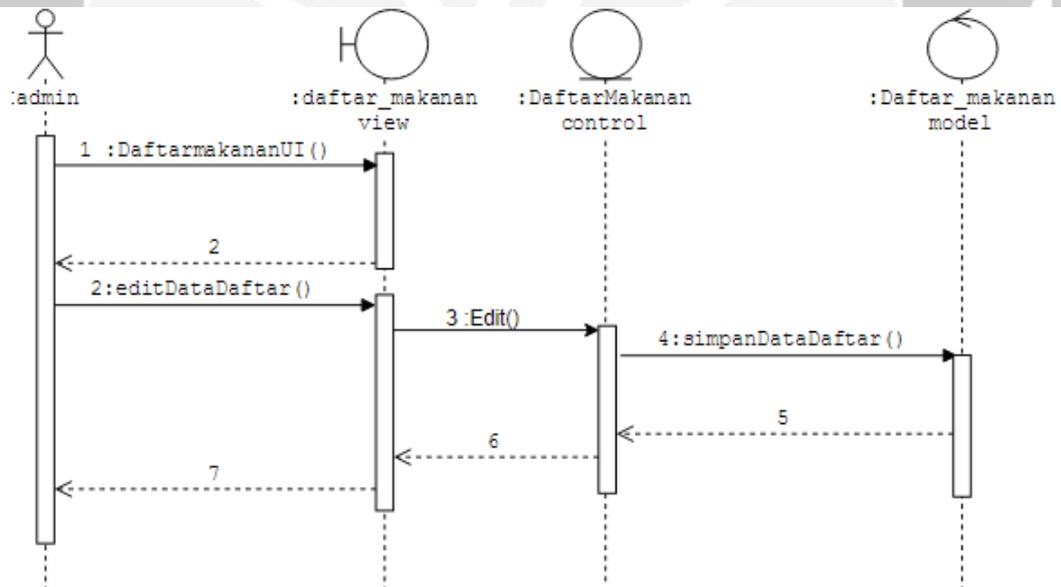
Gambar 2. 8 Sequence Diagram Hapus Data Kelompok Makanan

2.2.1.7 Fungsi Tambah Data Daftar Makanan



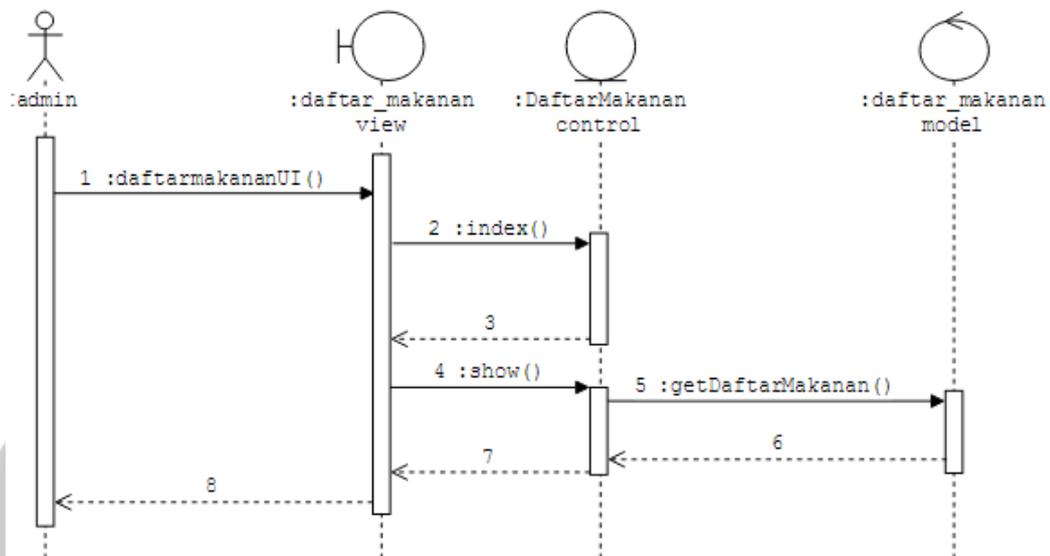
Gambar 2. 9 Sequence Diagram Tambah Data Daftar Makanan

2.2.1.8 Fungsi Edit Data Daftar Makanan



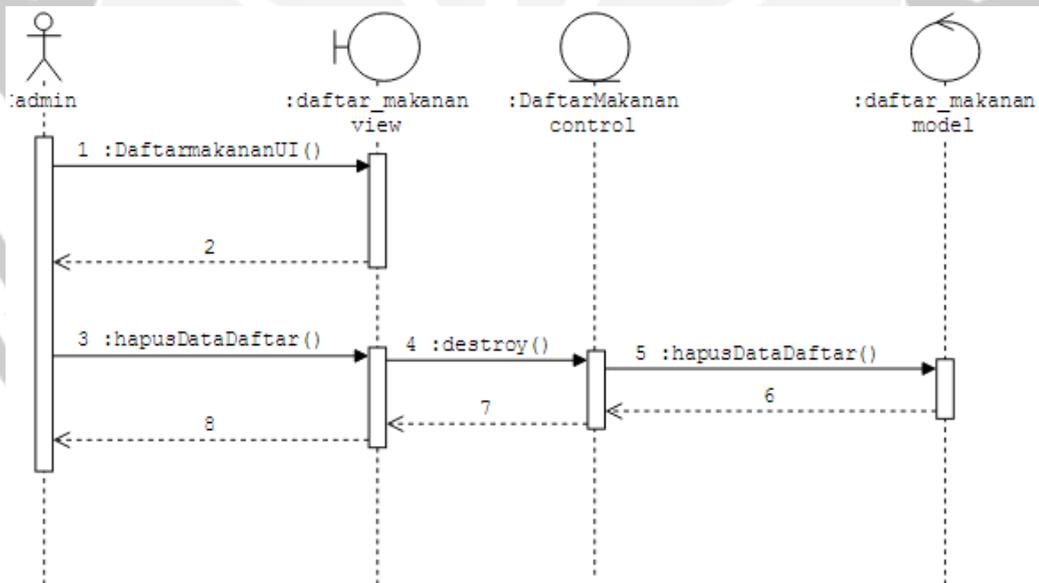
Gambar 2. 10 Sequence Diagram Edit Data Daftar Makanan

2.2.1.9 Fungsi Tampil Data Daftar Makanan



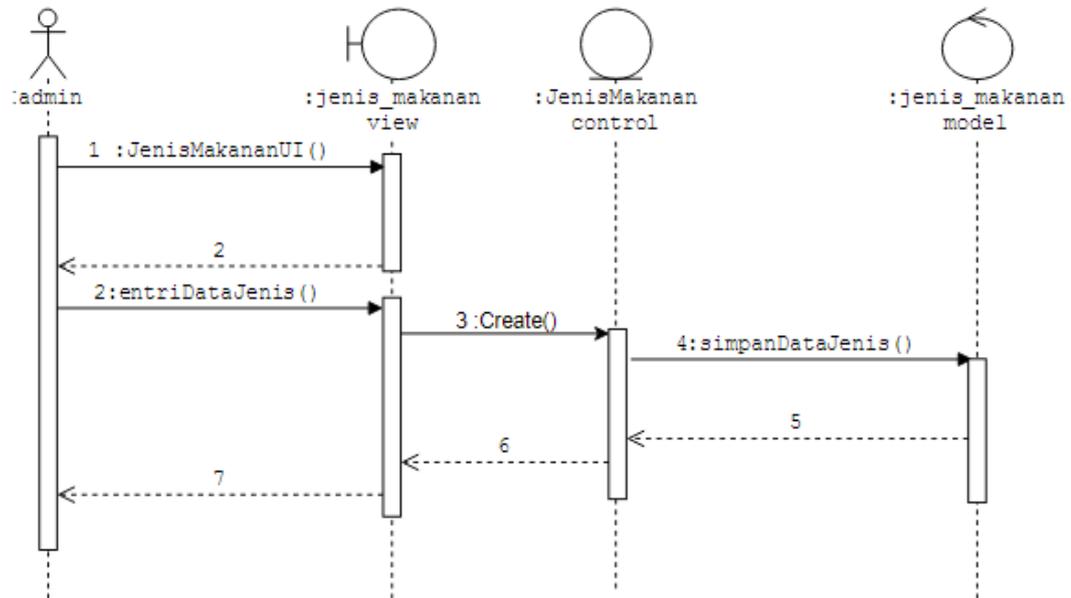
Gambar 2. 11 Sequence Diagram Tampil Data Daftar Makanan

2.2.1.10 Fungsi Hapus Data Daftar Makanan



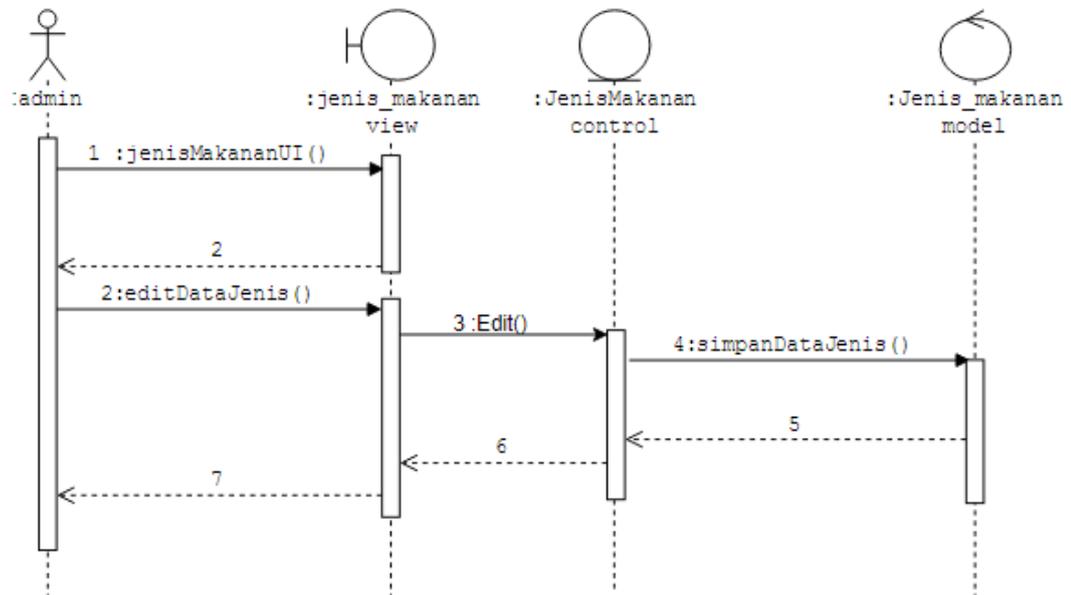
Gambar 2. 12 Sequence Diagram Hapus Data Daftar Makanan

2.2.1.11 Fungsi Tambah Data Jenis Makanan



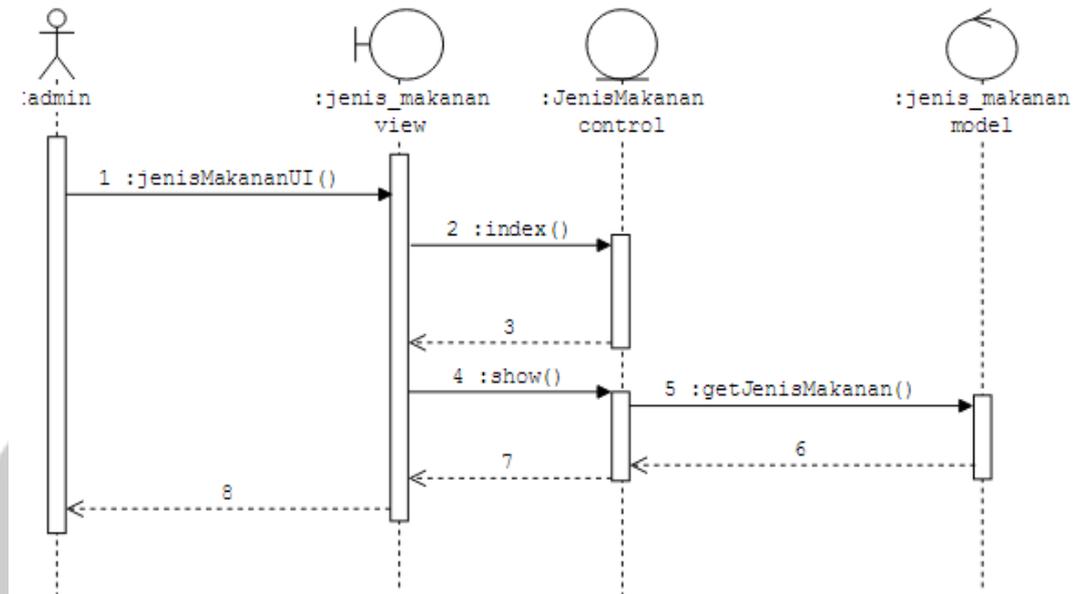
Gambar 2. 13 Sequence Diagram Tambah Data Jenis Makanan

2.2.1.12 Fungsi Edit Data Jenis Makanan



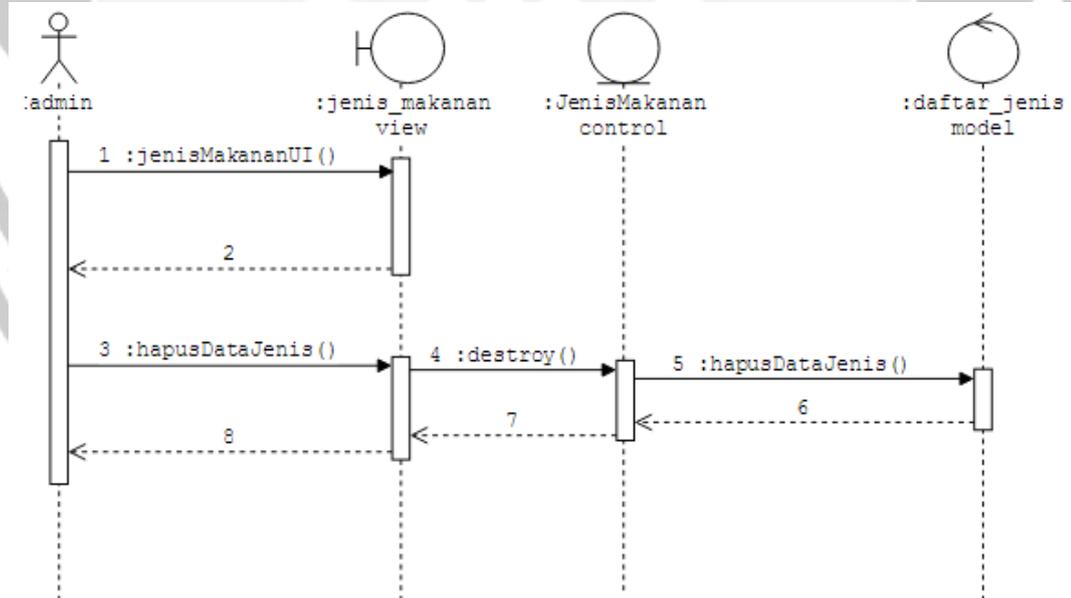
Gambar 2. 14 Sequence Diagram Edit Data Jenis Makanan

2.2.1.13 Fungsi Tampil Data Jenis Makanan



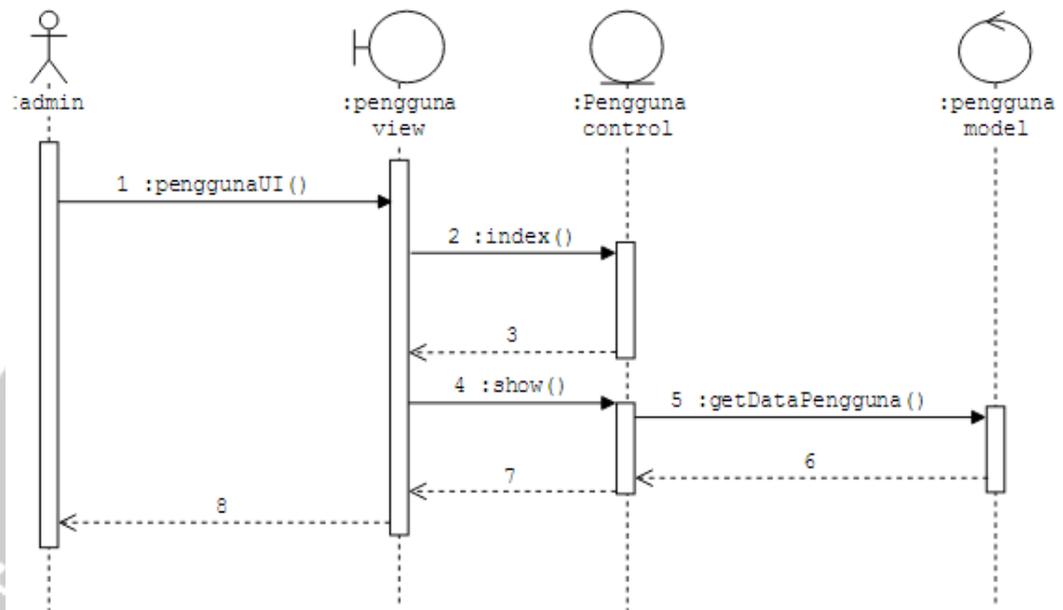
Gambar 2. 15 Sequence Diagram Tampil Data Jenis Makanan

2.2.1.14 Fungsi Hapus Data Jenis Makanan



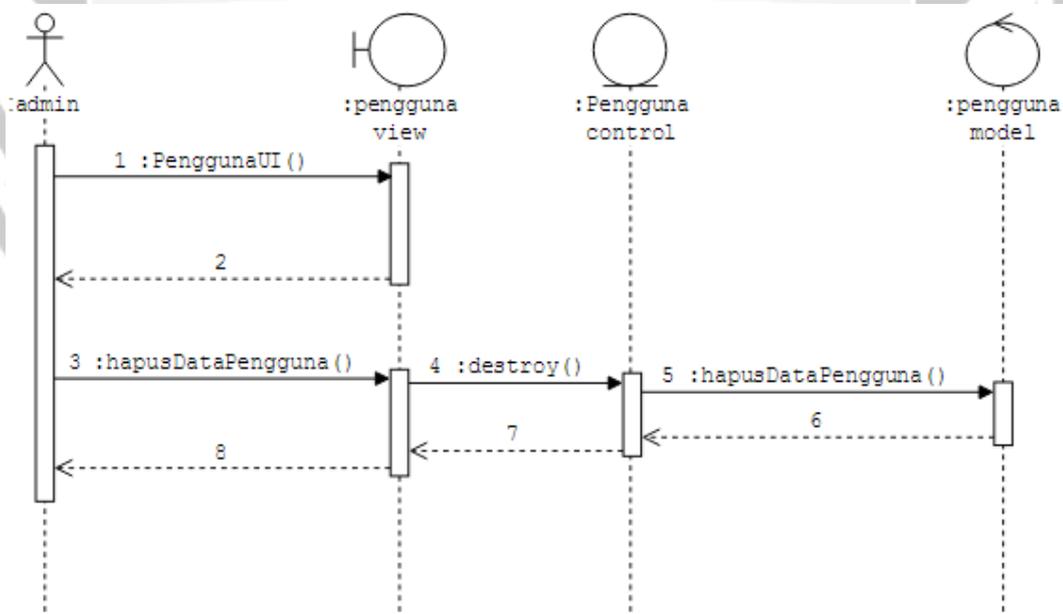
Gambar 2. 16 Sequence Diagram Hapus Data Jenis Makanan

2.2.1.15 Fungsi Tampil Data Pengguna



Gambar 2. 17 Sequence Diagram Tampil Data Pengguna

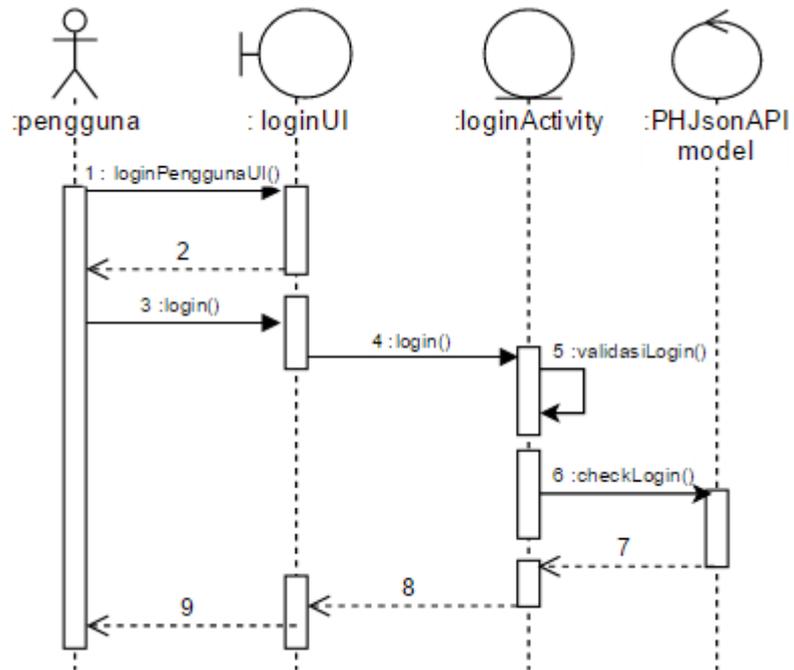
2.2.1.16 Fungsi Hapus Data Pengguna



Gambar 2. 18 Sequence Diagram Hapus Data Pengguna

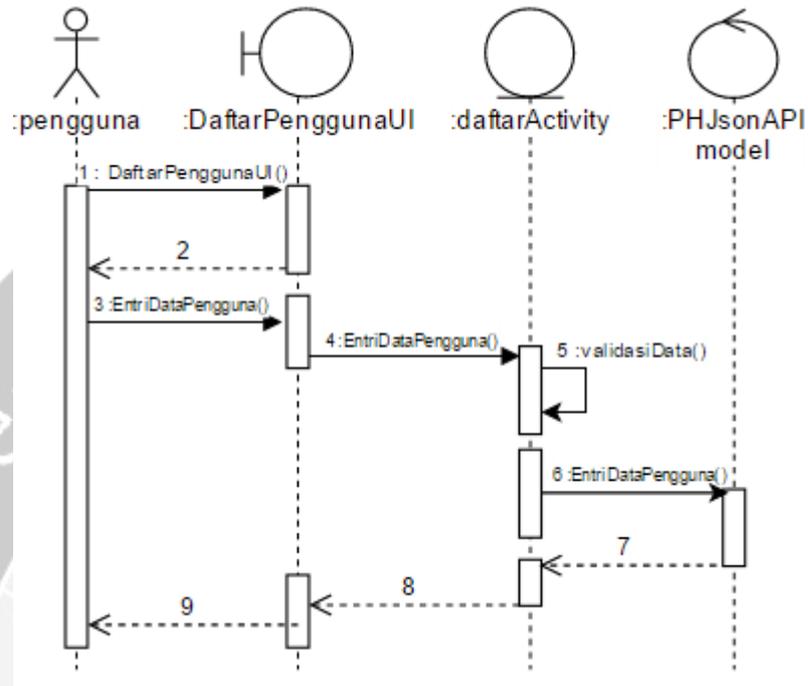
2.2.2 Sequence Diagram Mobile Pocket Healthy

2.2.2.1 Fungsi Login Pengguna



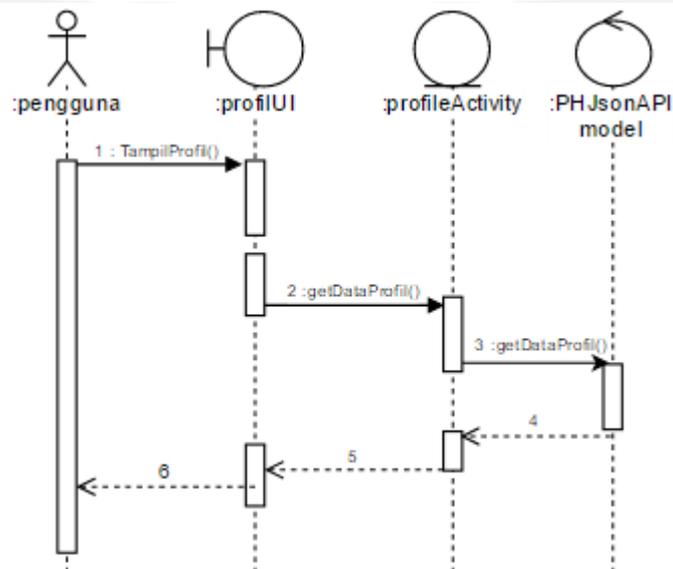
Gambar 2. 17 Sequence Diagram Fungsi Login Pengguna

2.2.2.2 Fungsi Daftar Pengguna



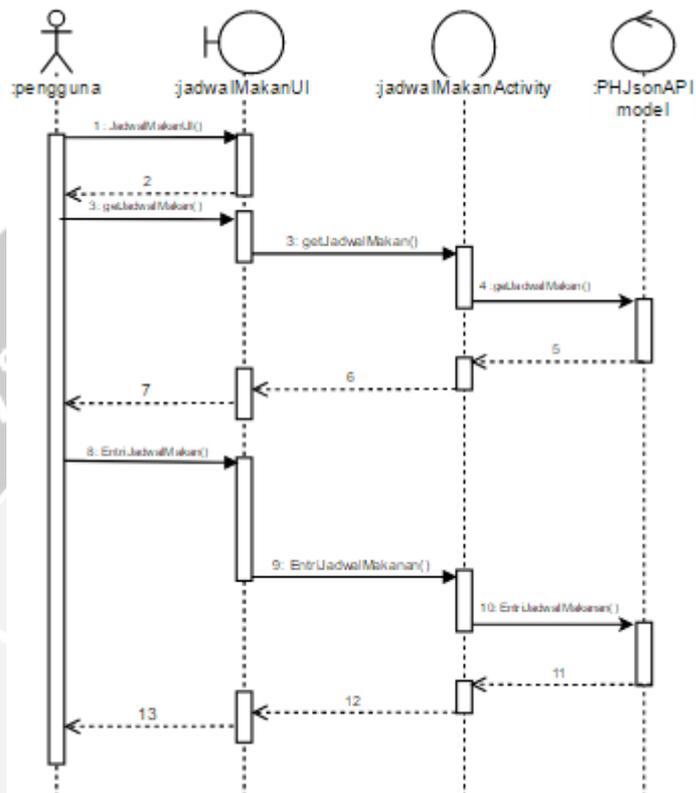
Gambar 2. 18 Sequence Diagram Fungsi Daftar Pengguna

2.2.2.3 Fungsi Tampil Profil



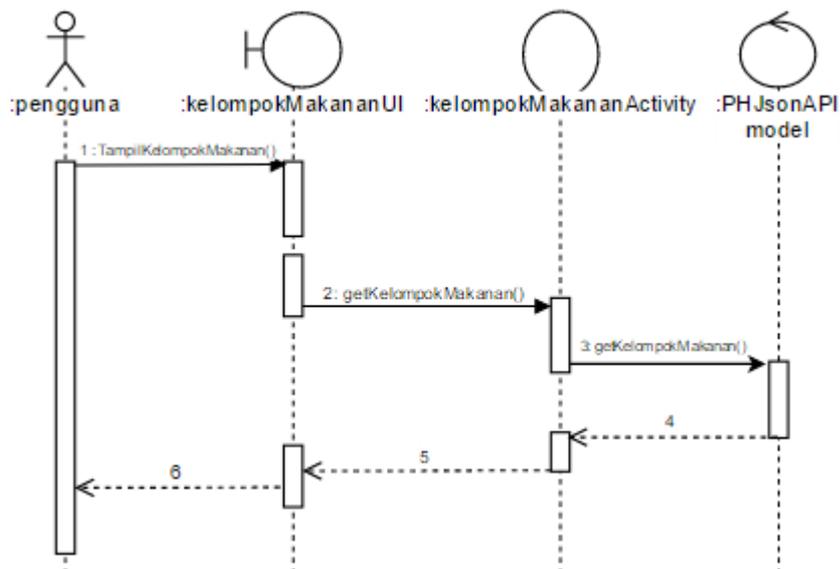
Gambar 2. 18 Sequence Diagram Fungsi Tampil Profil

2.2.2.4 Fungsi Jadwal Makan



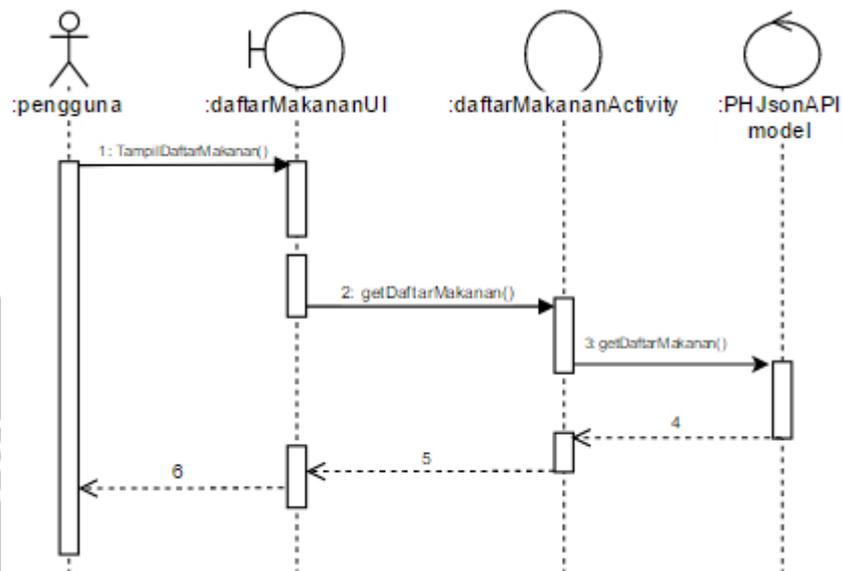
Gambar 2. 19 Sequence Diagram Fungsi Jadwal Makan

2.2.2.4 Fungsi Tampil Kelompok Makanan



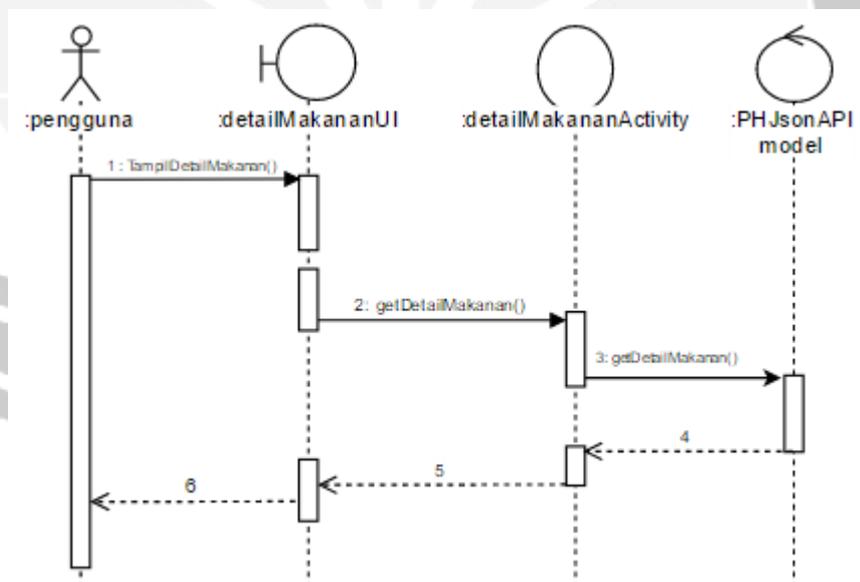
Gambar 2. 20 Sequence Diagram Fungsi Tampil Kelompok Makanan

2.2.2.5 Fungsi Tampil Daftar Makanan



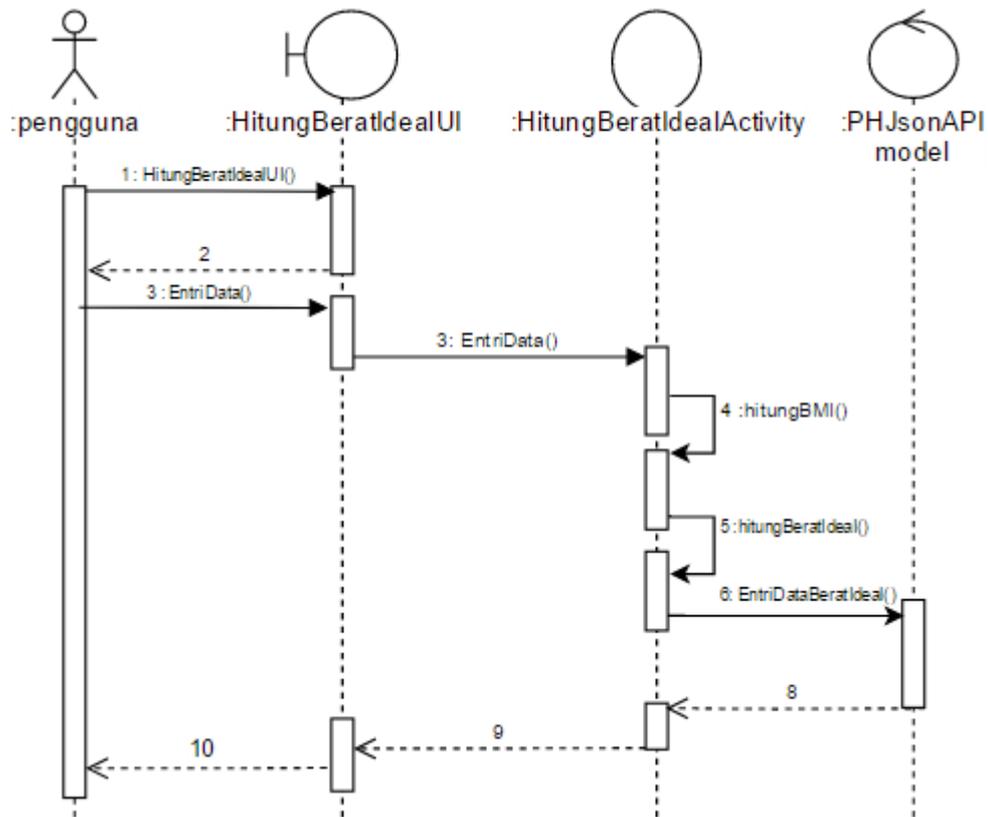
Gambar 2. 21 Sequence Diagram Fungsi Tampil Daftar Makanan

2.2.2.6 Fungsi Tampil Detail Makanan



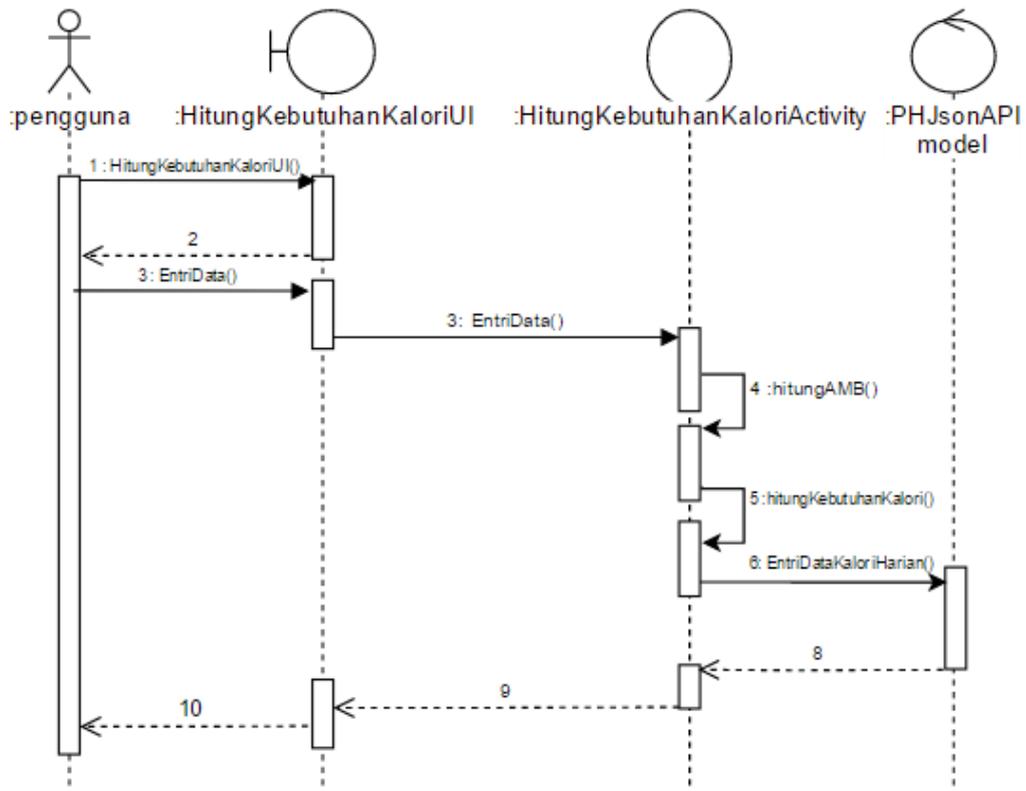
Gambar 2. 22 Sequence Diagram Fungsi Tampil Detail Makanan

2.2.2.7 Fungsi Hitung Berat Ideal



Gambar 2. 23 Sequence Diagram Fungsi Hitung Berat Ideal

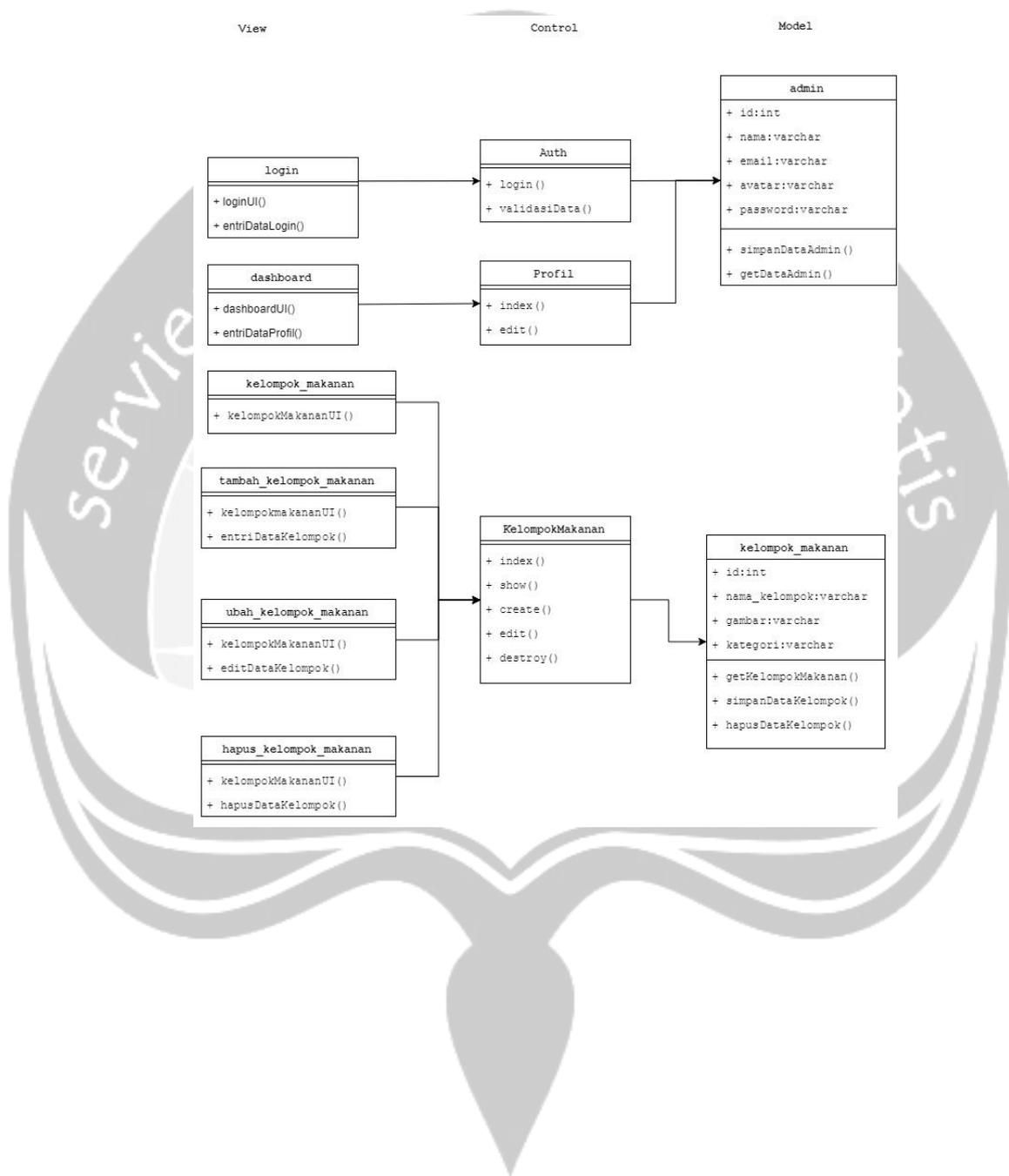
2.2.2.8 Fungsi Hitung Kebutuhan Kalori

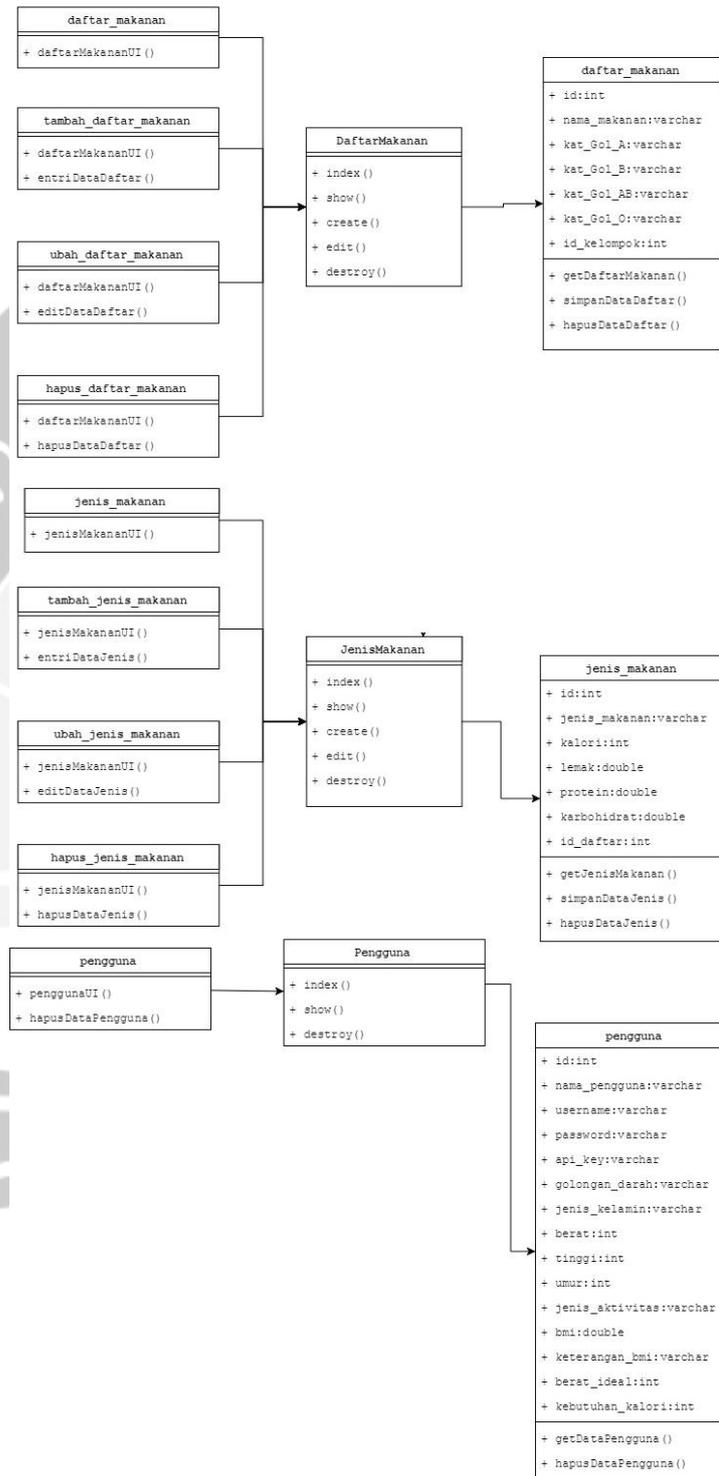


Gambar 2. 24 Sequence Diagram Fungsi Hitung Kebutuhan Kalori

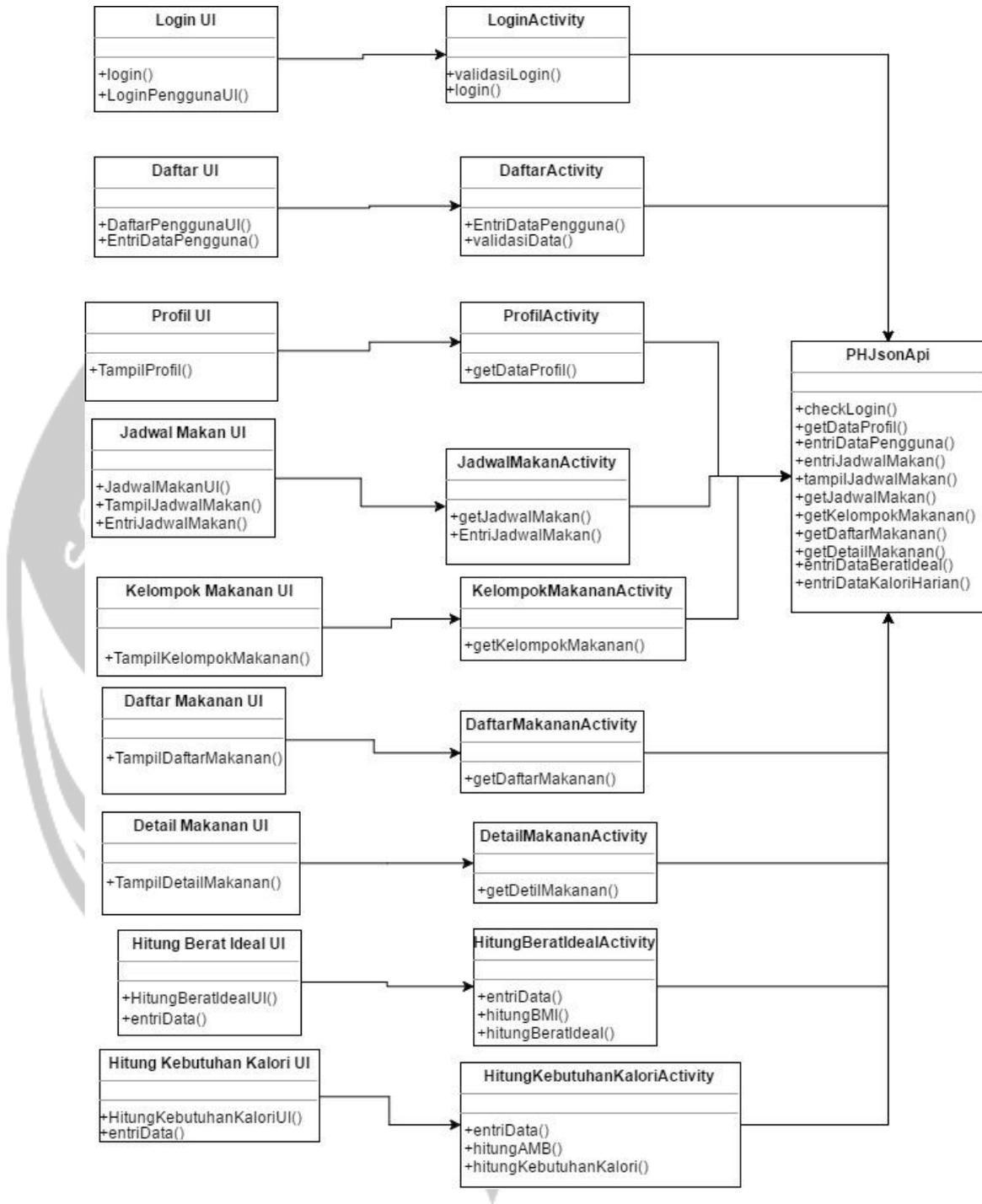
2.2.3 Class Diagram

2.2.3.1 Class Diagram Web PH





2.2.3.2 Class Diagram Pocket Healthy



Gambar 2. 26 Class Diagram Pocket Healthy

2.2.4 Class Diagram Specific Descriptions Web PH

2.2.4.2 Specific Design Class login

login	<<boundary>>
+ loginUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini dan juga menampilkan form login.	
+ entriDataAdmin() Operasi ini digunakan untuk menginputkan data username dan password admin untuk login.	

2.2.4.4 Specific Design Class dashboard

dashboard	<<boundary>>
+ dashboardUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ entriDataProfil () Operasi ini digunakan untuk menginputkan data profil admin untuk diupdate.	

2.2.4.5 Specific Design Class kelompok_makanan

kelompok_makanan	<<boundary>>
+ kelompokmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

2.2.4.6 Specific Design Class tambah_kelompok_makanan

tambah_kelompok_makanan	<<boundary>>
+ kelompokmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ entriDataKelompok() Operasi ini digunakan untuk menginputkan data kelompok makanan.	

2.2.4.7 Specific Design Class ubah_kelompok_makanan

ubah_kelompok_makanan	<<boundary>>
+ kelompokmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ editDataKelompok() Operasi ini digunakan untuk menginputkan data ubah kelompok makanan.	

2.2.4.8 Specific Design Class hapus_kelompok_makanan

hapus_kelompok_makanan	<<boundary>>
+ kelompokmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+ hapusDataKelompok() Operasi ini digunakan untuk menghapus data kelompok makanan.	

2.2.4.9 Specific Design Class daftar_makanan

daftar_makanan	<<boundary>>
+ daftarmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

2.2.4.10 Specific Design Class tambah_daftar_makanan

daftar_makanan	<<boundary>>
+ daftarmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + entriDataDaftar() Operasi ini digunakan untuk menginputkan data daftar makanan.	

2.2.4.11 Specific Design Class ubah_daftar_makanan

daftar_makanan	<<boundary>>
+ daftarmakananUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + editDataDaftar() Operasi ini digunakan untuk menginputkan data ubah daftar makanan.	

2.2.4.12 Specific Design Class hapus_daftar_makanan

Hapus_daftar_makanan	<<boundary>>

<p>+ daftarmakananUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+ hapusDataDaftar()</p> <p>Operasi ini digunakan untuk menghapus data daftar makanan.</p>
--

2.2.4.13 Specific Design Class jenis_makanan

daftar_makanan	<<boundary>>
<p>+ jenismakananUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p>	

2.2.4.14 Specific Design Class tambah_jenis_makanan

jenis_makanan	<<boundary>>
<p>+ daftarmakananUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+ entriDataDaftar()</p> <p>Operasi ini digunakan untuk menginputkan data daftar makanan.</p>	

2.2.4.15 Specific Design Class ubah_jenis_makanan

Ubah_jenis_makanan	<<boundary>>
<p>+ daftarmakananUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p>	

<pre>+ editDataJenis()</pre> <p>Operasi ini digunakan untuk menginputkan data ubah jenis makanan.</p>

2.2.4.16 Specific Design Class hapus_jenis_makanan

hapus_jenis_makanan	<<boundary>>
<pre>+ daftarmakananUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <pre>+ hapusDataJenis()</pre> <p>Operasi ini digunakan untuk menghapus data jenis makanan.</p>	

2.2.4.17 Specific Design Class pengguna

pengguna	<<boundary>>
<pre>+ PenggunaUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <pre>+ hapusPengguna()</pre> <p>Operasi ini digunakan untuk menghapus data pengguna</p>	

2.2.4.18 Specific Design Class Auth

Auth	<<control>>
<pre>+ validasiData()</pre> <p>Operasi ini digunakan untuk mengecek inputan valid atau tidak.</p> <pre>+login ()</pre>	

Operasi ini digunakan untuk masuk ke dalam system.

2.2.4.19 Specific Design Class Profil

profil_control	<<control>>
+index() Operasi ini digunakan untuk mengeset view dan mengecek session login.	
+edit() Operasi ini digunakan untuk mengubah data profil admin.	

2.2.4.20 Specific Design Class KelompokMakanan

KelompokMakanan	<<control>>
+index() Operasi ini digunakan untuk mengeset view dan mengecek session login.	
+ show() Operasi ini digunakan untuk menampilkan data kelompok makanan.	
+ create() Operasi ini digunakan untuk menyimpan data kelompok makanan yang ditambahkan.	
+ edit() Operasi ini digunakan untuk menyimpan perubahan data kelompok makanan berdasarkan id.	
+ destroy() Operasi ini digunakan untuk menghapus data kelompok makanan berdasarkan id.	

2.2.4.21 Specific Design Class DaftarMakanan

DaftarMakanan	<<control>>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +create() Operasi ini digunakan untuk menyimpan data daftar makanan yang ditambahkan. +show() Operasi ini digunakan untuk menampilkan data daftar makanan. +edit() Operasi ini digunakan untuk menyimpan perubahan data daftar makanan berdasarkan id. +destroy() Operasi ini digunakan untuk menghapus data daftar makanan berdasarkan id.</pre>	

2.2.4.21 Specific Design Class JenisMakanan

DaftarMakanan	<<control>>
<pre>+index() Operasi ini digunakan untuk mengeset view dan mengecek session login. +create() Operasi ini digunakan untuk menyimpan data jenis makanan yang ditambahkan. +show()</pre>	

Operasi ini digunakan untuk menampilkan data jenis makanan.

+ edit()

Operasi ini digunakan untuk menyimpan perubahan data jenis makanan berdasarkan id.

+ destroy()

Operasi ini digunakan untuk menghapus data jenis makanan berdasarkan id.

2.2.4.19 Specific Design Class pengguna_control

pengguna_control	<<control>>
<p>+index()</p> <p>Operasi ini digunakan untuk mengeset view dan mengecek session login.</p> <p>+ show()</p> <p>Operasi ini digunakan untuk menampilkan data pengguna aplikasi mobile.</p> <p>+ destroy()</p> <p>Operasi ini digunakan untuk menghapus data pengguna aplikasi mobile.</p>	

2.2.4.20 Specific Design Class admin

admin	<<entity>>
<p>+ id : int</p> <p>Atribut yang digunakan untuk menyimpan id admin.</p> <p>+ name : varchar</p> <p>Atribut yang digunakan untuk menyimpan nama admin.</p> <p>+ email : varchar</p>	

Atribut yang digunakan untuk menyimpan email.
+ password : varchar
Atribut yang digunakan untuk menyimpan password admin.
+ avatar: varchar
Atribut yang digunakan untuk menyimpan avatar admin.

+ simpanDataAdmin()
Operasi ini digunakan untuk menyimpan data admin.
+ getDataAdmin()
Operasi ini digunakan untuk mendapatkan data admin.

2.2.4.22 Specific Design Class kelompok_makanan

kelompok_makanan	<<entity>>
+ id : int Atribut yang digunakan untuk menyimpan id kelompok makanan. + nama_kelompok : varchar Atribut yang digunakan untuk menyimpan nama kelompok makanan. + gambar : varchar Atribut yang digunakan untuk menyimpan gambar kelompok makanan. + kategori: varchar Atribut yang digunakan untuk menyimpan kategori.	
+ getDataKelompokMakanan() Operasi ini digunakan untuk mendapatkan data kelompok makanan. + simpanDataKelompokMakanan() Operasi ini digunakan untuk menyimpan data kelompok makanan.	

```
+ hapusDataKelompokMakanan()
```

Operasi ini digunakan untuk menghapus data kelompok makanan.

2.2.4.23 Specific Design Class daftar_makanan

daftar_makanan	<<entity>>
<pre>+ id : int</pre> <p>Atribut yang digunakan untuk menyimpan id daftar makanan.</p>	
<pre>+ id_kelompok : int</pre> <p>Atribut yang digunakan untuk menyimpan id kelompok makanan dan merupakan foreign key dari kelompok makanan.</p>	
<pre>+ nama_makanan: varchar</pre> <p>Atribut yang digunakan untuk menyimpan nama daftar makanan.</p>	
<pre>+ kat_Gol_A : varchar</pre> <p>Atribut yang digunakan untuk menyimpan kategori makanan golongan darah A.</p>	
<pre>+ kat_Gol_B : varchar</pre> <p>Atribut yang digunakan untuk menyimpan kategori makanan golongan darah B.</p>	
<pre>+ kat_Gol_AB : varchar</pre> <p>Atribut yang digunakan untuk menyimpan kategori makanan golongan darah AB.</p>	
<pre>+ kat_Gol_o : varchar</pre> <p>Atribut yang digunakan untuk menyimpan kategori makanan golongan darah O.</p>	
<pre>+ getDaftarMakanan()</pre>	

Operasi ini digunakan untuk mendapatkan data daftar makanan.

+ simpanDataDaftar ()

Operasi ini digunakan untuk menyimpan data daftar makanan.

+ hapusDataDaftar ()

Operasi ini digunakan untuk menghapus data daftar makanan.

2.2.4.24 Specific Design Class jenis_makanan

jenis_makanan	<<entity>>
<p>+ id : int Atribut yang digunakan untuk menyimpan id jenis makanan.</p>	
<p>+ id_daftar : int Atribut yang digunakan untuk menyimpan id daftar makanan dan merupakan foreign key dari daftar makanan.</p>	
<p>+ jenis_makanan: varchar Atribut yang digunakan untuk menyimpan nama jenis.</p>	
<p>+ porsi : varchar Atribut yang digunakan untuk menyimpan porsi makanan.</p>	
<p>+ kalori : varchar Atribut yang digunakan untuk menyimpan kalori makanan.</p>	
<p>+ lemak : double Atribut yang digunakan untuk menyimpan lemak makanan.</p>	
<p>+ karbohidrat : double Atribut yang digunakan untuk menyimpan karbohidrat makanan.</p>	

+ protein : double

Atribut yang digunakan untuk menyimpan protein makanan.

+ getJenisMakanan()

Operasi ini digunakan untuk mendapatkan data jenis makanan.

+ simpanDataJenis()

Operasi ini digunakan untuk menyimpan data jenis makanan.

+ hapusDataJenis()

Operasi ini digunakan untuk menghapus data jenis makanan.

2.2.4.12 Specific Design Class pengguna

pengguna_model

<<entity>>

+ id: int

Atribut yang digunakan untuk menyimpan id pengguna.

+ nama : varchar

Atribut yang digunakan untuk menyimpan nama pengguna.

+ username : varchar

Atribut yang digunakan untuk menyimpan username.

+ password : varchar

Atribut yang digunakan untuk menyimpan password pengguna.

+ api_key : varchar

Atribut yang digunakan untuk menyimpan api key pengguna.

+ golongan_darah : varchar

Atribut yang digunakan untuk menyimpan golongan darah pengguna.

+ jenis_kelamin : varchar

Atribut yang digunakan untuk menyimpan jenis kelamin pengguna.

+ berat : int

Atribut yang digunakan untuk menyimpan berat badan pengguna.

+ tinggi : int

Atribut yang digunakan untuk menyimpan tinggi badan pengguna.

+ umur : int

Atribut yang digunakan untuk menyimpan umur pengguna.

+ jenis_aktivitas : varchar

Atribut yang digunakan untuk menyimpan jenis aktivitas pengguna.

+ bmi : double

Atribut yang digunakan untuk menyimpan nilai indeks massa tubuh pengguna.

+ keterangan_bmi: varchar

Atribut yang digunakan untuk menyimpan keterangan bmi berdasarkan nilai yang didapatkan.

+ berat_ideal : int

Atribut yang digunakan untuk menyimpan berat ideal pengguna.

+ total_kalori : int

Atribut yang digunakan untuk menyimpan jumlah kalori pengguna.

+ getDataPengguna ()

Operasi ini digunakan untuk mendapatkan data pengguna.

+ hapusDataPengguna()

Operasi ini digunakan untuk menghapus data pengguna.

2.2.5 Class Diagram Specific Descriptions Mobile PH

2.2.5.1 Specific Design Class Login UI

LoginUI	<<boundary>>
<p>+ LoginPenggunaUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+login()</p> <p>Operasi ini digunakan untuk mengambil data login yang diinputkan oleh user, yaitu login username dan password.</p>	

2.2.5.2 Specific Design Class Daftar UI

DaftarUI	<<boundary>>
<p>+ DaftarPenggunaUI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntriDataPengguna()</p> <p>Operasi ini digunakan untuk mengambil data daftar pengguna dan memasukannya kedalam database.</p>	

2.2.5.3 Specific Design Class Profil UI

TampilProfilUI	<<boundary>>
-----------------------	---------------------------------

+ TampilProfil () Operasi ini digunakan untuk menampilkan informasi pribadi pengguna/pasien.

2.2.5.4 Specific Design Class Jadwal Makan UI

JadwalMakanUI	<<boundary>>
+ JadwalMakanUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+TampilJadwalMakan() Operasi ini digunakan untuk menampilkan data jadwal makan.	
+EntriJadwalMakan() Operasi ini digunakan untuk mengambil data jadwal makan untuk dimasukkan kedalam database.	

2.2.5.5 Specific Design Class Kelompok Makanan UI

TampilKelompokMakananUI	<<boundary>>
+ TampilKelompokMakanan () Operasi ini digunakan untuk menampilkan semua data kelompok makanan yang ada didalam database.	

2.2.5.6 Specific Design Class Daftar Makanan UI

TampilDaftarMakananUI	<<boundary>>

Program Studi Teknik Informatika	DPPL – PH	45/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

<p>+ TampilDaftarMakananUI ()</p> <p>Operasi ini digunakan untuk menampilkan semua data daftar makanan dari kelompok makanan tertentu yang ada didalam database.</p>
--

2.2.5.7 Specific Design Class Detail Makanan UI

TampilDetailMakananUI	<<boundary>>
<p>+ TampilDetailMakananUI ()</p> <p>Operasi ini digunakan untuk menampilkan detail makanan dari daftar makanan yang ada didalam database.</p>	

2.2.5.8 Specific Design Class Hitung Berat Ideal UI

HitungBeratIdealUI	<<boundary>>
<p>+ HitungBeratIdealUI ()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntriData()</p> <p>Operasi ini digunakan untuk mengambil data untuk dihitung dan dimasukkan kedalam database.</p>	

2.2.5.8 Specific Design Class Hitung Kebutuhan Kalori UI

HitungKaloriHarianUI	<<boundary>>
<p>+ HitungKaloriHarianUI ()</p>	

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+EntriData()

Operasi ini digunakan untuk mengambil data untuk dihitung dan dimasukkan kedalam database.

2.2.5.9 Specific Design Class Login Activity

LoginActivity	<< Control >>
<p>+validasiLogin() Operasi ini digunakan untuk mencocokkan data inputan user dengan database.</p> <p>+login() Operasi ini digunakan untuk meneruskan hasil pencocokan data inputan ke database.</p>	

2.2.5.10 Specific Design Class Daftar Activity

DaftarActivity	<<control>>
<p>+EntriDataPengguna() Operasi ini digunakan untuk masukkan data daftar ke dalam database.</p> <p>+validasiData() Operasi ini digunakan untuk mencocokkan data inputan pengguna sudah benar atau belum dan apakah sudah terdaftar di database atau belum.</p>	

2.2.5.11 Specific Design Class Profil Activity

ProfilActivity	<< Control >>
-----------------------	----------------------------------

+getDataProfil() Operasi ini digunakan untuk mengambil data informasi pribadi/profil pasien dari database.

2.2.5.12 Specific Design Class Jadwal Makan Activity

JadwalMakanActivity	<< Control >>
+getJadwalMakan() Operasi ini digunakan untuk mengambil data jadwal makan tertentu.	
+EntriJadwalMakan() Operasi ini digunakan untuk memasukkan jadwal makan yang telah dibuat kedalam database.	

2.2.5.13 Specific Design Class Kelompok Makanan Activity

KelompokMakananActivity	<< Control >>
+getKelompokMakanan() Operasi ini digunakan untuk mengambil data kelompok makanan dari database.	

2.2.5.14 Specific Design Class Daftar Makanan Activity

DaftarMakananActivity	<< Control >>
+getKelompokMakanan() Operasi ini digunakan untuk mengambil data daftar makanan dari database.	

2.2.5.15 Specific Design Class Detail Makanan Activity

DetailMakananActivity	<< Control >>
<code>+getKelompokMakanan()</code> Operasi ini digunakan untuk mengambil data detail makanan dari database.	

2.2.5.16 Specific Design Class Hitung Berat Ideal Activity

HitungBeratIdealActivity	<< Control >>
<code>+entriData()</code> Operasi ini digunakan untuk memasukan data berat tertentu kedalam database. <code>+hitungBMI()</code> Operasi ini digunakan untuk melakukan kalkulasi bmi berdasarkan data yang telah dimasukan. <code>+hitungBeratIdeal()</code> Operasi ini digunakan untuk melakukan kalkulasi berat ideal berdasarkan data yang telah dimasukan.	

2.2.5.17 Specific Design Class Hitung Kebutuhan Kalori Harian Activity

HitungBeratIdealActivity	<< Control >>
<code>+entriData()</code> Operasi ini digunakan untuk memasukan data kalori tertentu kedalam database. <code>+hitungAMB()</code>	

Operasi ini digunakan untuk melakukan kalkulasi AMB berdasarkan data yang telah dimasukan.

+hitungKebutuhanKalori()

Operasi ini digunakan untuk melakukan kalkulasi kebutuhan kalori berdasarkan data yang telah dimasukan.

2.2.5.18 Specific Design Class PHJsonApi

PHJsonApi	<< Entity >>
<p>+ checkLogin() Operasi ini digunakan untuk melakukan pengecekan login pengguna dengan data yang ada di database.</p> <p>+getDataProfil() Operasi ini digunakan untuk mendapatkan data informasi pribadi dan mengubahnya kedalam json.</p> <p>+entriDataPengguna() Operasi ini digunakan untuk mendapatkan data pengguna dan memasukkannya ke dalam database.</p> <p>+tampilJadwalMakan() Operasi ini digunakan untuk mendapatkan data jadwal makan dan mengubahnya kedalam json.</p> <p>+getKelompokMakanan() Operasi ini digunakan untuk mendapatkan data kelompok makanan dan mengubahnya kedalam json.</p> <p>+getDaftarMakanan() Operasi ini digunakan untuk mendapatkan data daftar makanan dan mengubahnya kedalam json.</p> <p>+getDetailMakanan() Operasi ini digunakan untuk mendapatkan data detail makanan dan mengubahnya kedalam json.</p>	

Program Studi Teknik Informatika	DPPL – PH	50/ 86
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```
+entriDataBeratIdeal()
```

Operasi ini digunakan untuk mendapatkan data berat ideal dan memasukkannya ke dalam database.

```
+entriDataKaloriHarian()
```

Operasi ini digunakan untuk mendapatkan data kalori harian dan memasukkannya ke dalam database.



3. Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas admin			
Nama	Tipe	Panjang	Keterangan
Id	INT	10	Id user, primary key
nama	VARCHAR	255	Nama admin
username	VARCHAR	255	Username
email	VARCHAR	255	Email admin
avatar	VARCHAR	255	Avatar admin
password	VARCHAR	255	Password admin

3.1.2 Deskripsi Entitas kelompok_makanan			
Nama	Tipe	Panjang	Keterangan
Id	INT	10	Id kelompok makanan, primary key
Nama_kelompok	VARCHAR	50	Nama kelompok makanan
gambar	VARHCAR	50	Gambar kelompok makanan
kategori	VARCHAR	50	Kategori kelompok makanan

3.1.3 Deskripsi Entitas daftar_makanan			
Nama	Tipe	Panjang	Keterangan
Id	INT	10	Id daftar makanan, primary key
Nama_makanan	VARCHAR	50	Nama makanan
kat_Gol_A	VARCHAR	50	Kategori golongan A
kat_Gol_B	VARCHAR	50	Kategori golongan B
kat_Gol_AB	VARCHAR	50	Kategori golongan AB
kat_Gol_O	VARCHAR	50	Kategori golongan O

Id_kelompok	INT	10	Id kelompok makanan, Foreign key dari entitas kelompok_makanan
-------------	-----	----	--

3.1.4 Deskripsi Entitas jenis_makanan

Nama	Type	Panjang	Keterangan
Id	INT	10	Id jenis makanan, primary key
Jenis_makanan	INT	10	Nama jenis makanan
Kalori	INT	11	Nilai kalori pada makanan
Lemak	Double	-	Nilai lemak pada makanan
Protein	Double	-	Nilai protein pada makanan
Karbohidrat	Double	-	Nilai karbohidrat pada makanan
Id_daftar	INT	11	Id daftar, foreign key dari entitas daftar_makanan

3.1.5 Deskripsi Entitas jadwal_makan

Nama	Type	Panjang	Keterangan
Id	INT	10	Id jadwal makan, primary key
Jenis_jadwal	VARCHAR	50	Jenis jadwal makan
tanggal	DATE	-	Tanggal jadwal
Id_pengguna	INT	11	Id pengguna, foreign key dari entitas pengguna

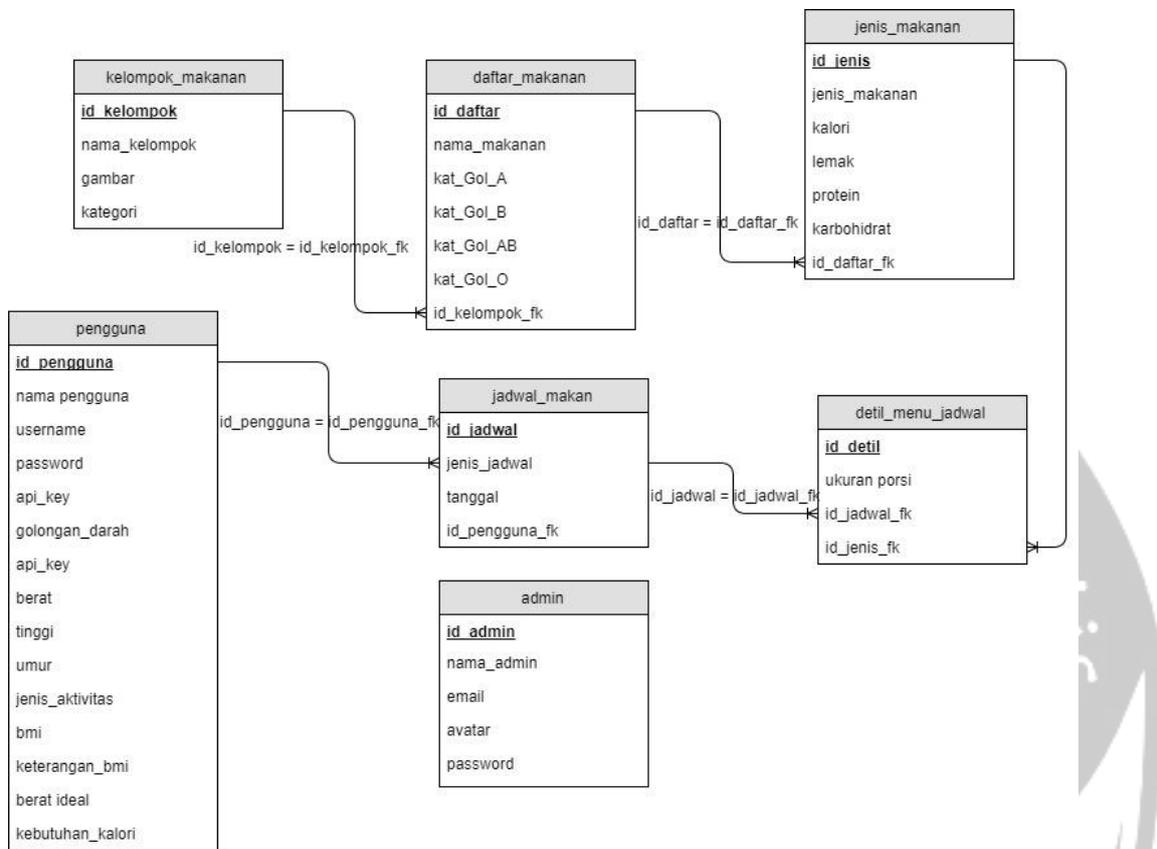
3.1.6 Deskripsi Entitas detail_menu_jadwal			
Nama	Tipe	Panjang	Keterangan
Id	INT	10	Id detail menu jadwal, primary key
Ukuran_porsi	INT	10	Ukuran porsi dari makanan
Id_jadwal	INT	10	Id jadwal, foreign key dari entitas jadwal_makan
Id_jenis	INT	10	Id jenis, foreign key dari entitas jenis_makanan

3.1.7 Deskripsi Entitas pengguna			
Nama	Tipe	Panjang	Keterangan
Id	INT	10	Id pengguna, primary key
Nama_pengguna	VARCHAR	255	Nama pengguna
Username	VARCHAR	255	Username
Password	VARCHAR	255	Password pengguna
Api_key	VARCHAR	32	Salt enkripsi password
Golongan_darah	VARCHAR	255	Golongan darah pengguna
Jenis_kelamin	VARCHAR	255	Jenis kelamin pengguna
Berat	INT	11	Berat pengguna
Tinggi	INT	11	Tinggi pengguna
Umur	INT	11	Umur pengguna
Jenis_aktivitas	VARCHAR	255	Jenis aktivitas pengguna
Bmi	DOUBLE	-	Nilai indeks massa tubuh
Keterangan_bmi	VARCHAR	50	Keterangan berdasarkan nilai indeks massa tubuh

Berat_ideal	INT	11	Berat ideal
Jumlah_kalori	INT	11	Jumlah kalori pengguna



3.1 Physical Data Model



Gambar 3.1 Physical Data Model

4 Deskripsi Perancangan Antarmuka

4.1.1. Sketsa UI Perangkat Lunak Web PH dan Deskripsinya

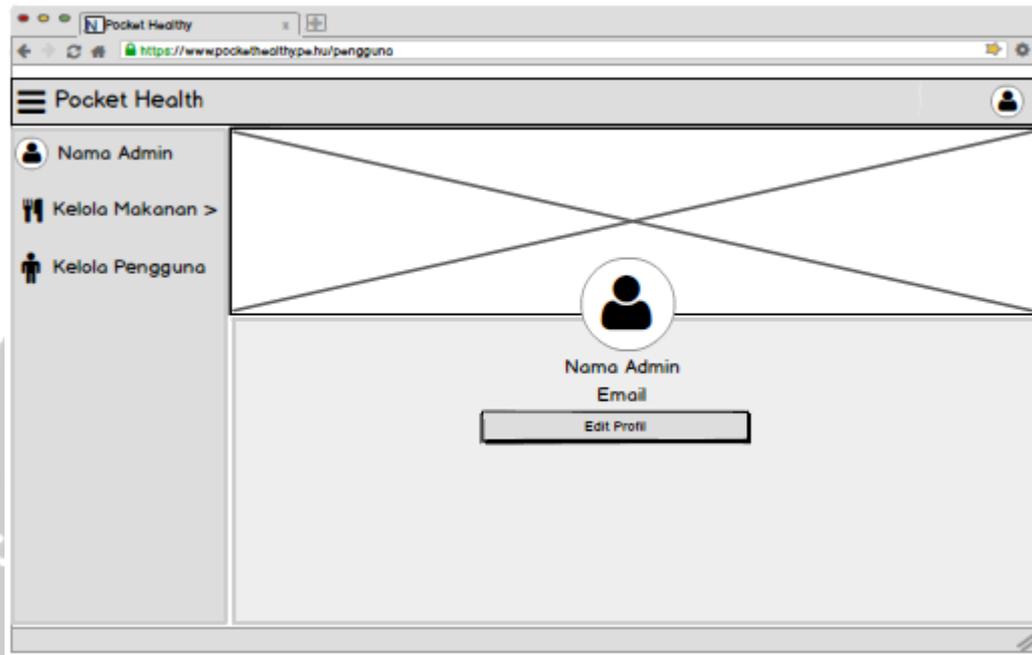
4.1.1.1. Antarmuka Login



Gambar 4. 1 Antarmuka Login

Antarmuka gambar 4.1 digunakan untuk melakukan proses login kedalam sistem aplikasi web. Untuk mendapatkan akses masuk kedalam sistem, pengguna harus memasukkan nama pengguna dan kata sandi terlebih dahulu pada textbox yang telah disediakan. Pada saat tombol masuk ditekan, sistem akan mengecek nama pengguna dan kata sandi yang telah dimasukan dengan data yang tersimpan di database benar atau tidak. Jika benar maka pengguna akan masuk kedalam sistem. Sedangkan sebaliknya jika tidak benar maka akan muncul pesan peringatan kesalahan. Apabila tombol masuk ditekan maka akan dilanjutkan ke antarmuka beranda.

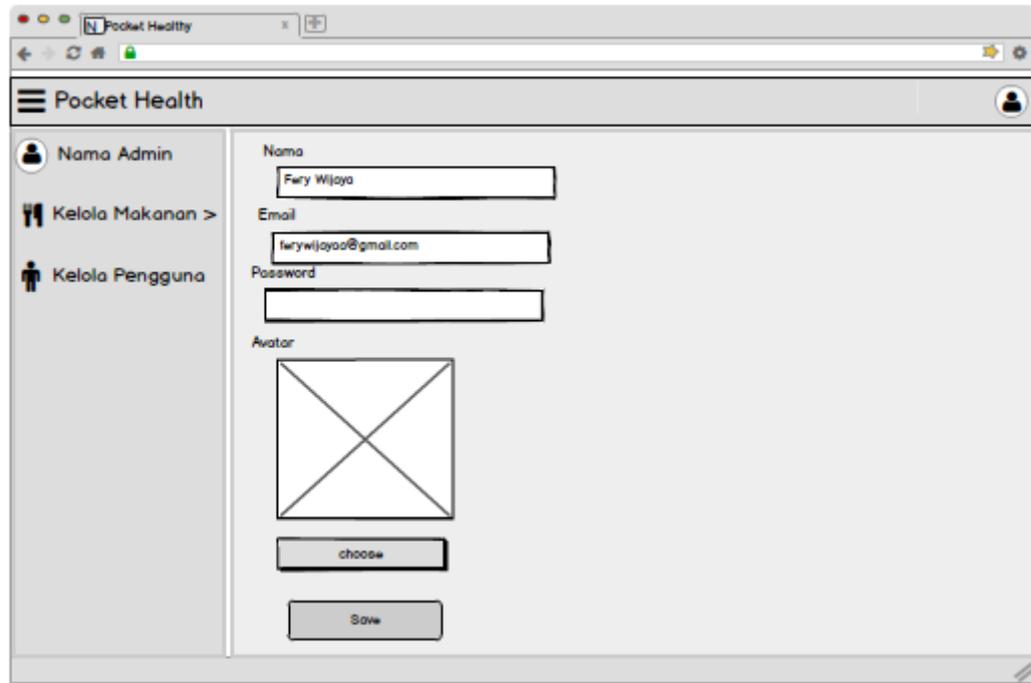
4.1.1.2. Antarmuka Beranda



Gambar 4. 2 Antarmuka Beranda

Antarmuka gambar 4.2 digunakan untuk menampilkan profil dari admin yang sedang login ke sistem. Apabila tombol Edit Profil ditekan maka akan dilanjutkan ke antarmuka Edit Profil.

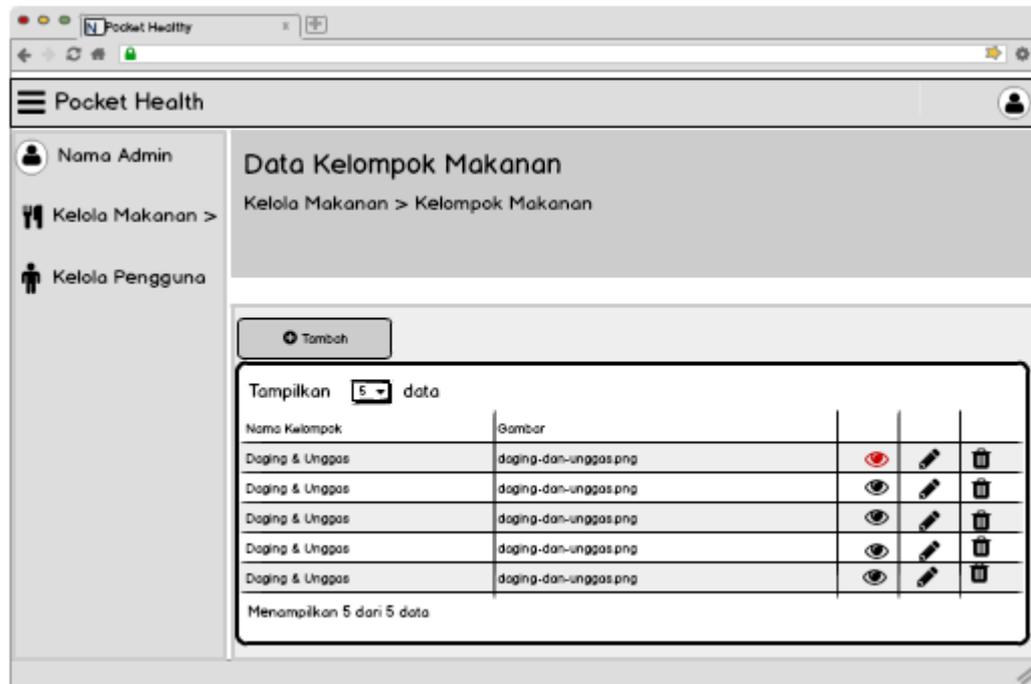
4.1.1.3. Antarmuka Edit Profil



Gambar 4. 3 Antarmuka Edit Profil

Antarmuka gambar 4.3 digunakan untuk mengedit profil admin seperti mengganti email ataupun password serta avatar yang digunakan.

4.1.1.4. Antarmuka Kelola Kelompok Makanan



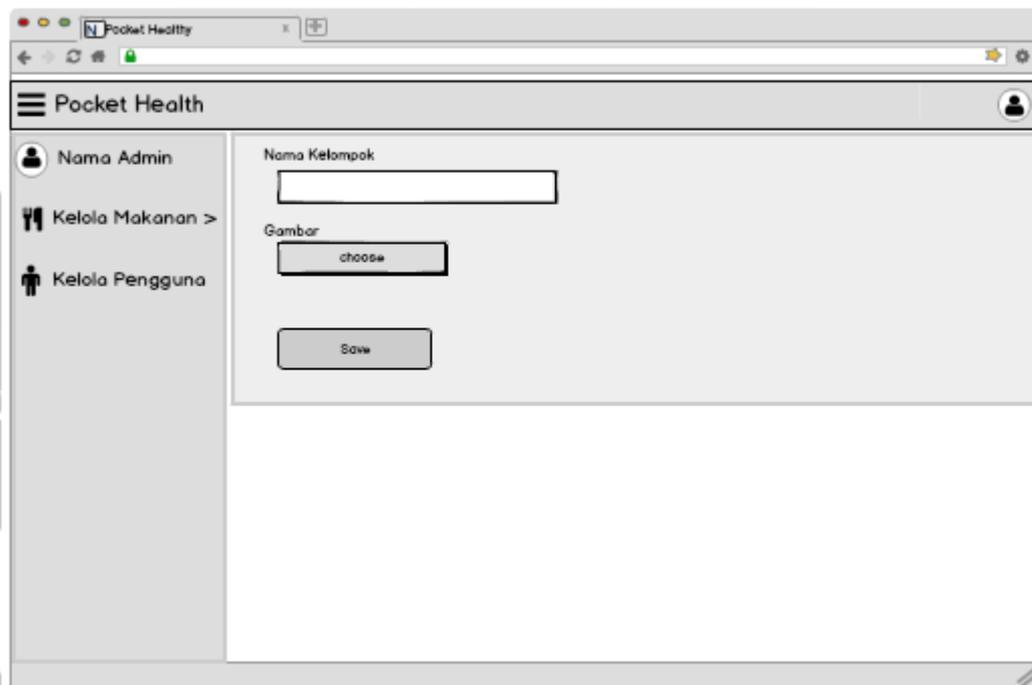
Gambar 4. 4 Antarmuka Kelola Kelompok Makanan

Antarmuka gambar 4.4 digunakan untuk mengelola data kelompok makanan. Pada antarmuka ini menampilkan data kelompok. Data ditampilkan perhalaman yang dapat diatur melalui dropdown untuk menampilkan data, dan terdapat kolom pencarian untuk mencari data tertentu. Data yang tampil juga dapat di urutkan berdasarkan huruf dari kecil ke besar atau sebaliknya.

Ada empat tombol dalam antarmuka ini, yang pertama adalah tombol tambah yang digunakan jika pengguna ingin menambahkan data kelompok makanan. Lalu tombol kedua adalah tombol tampil gambar yang berfungsi untuk menampilkan gambar dari data kelompok makanan yang tersimpan pada database. Lalu tombol ketiga adalah tombol ubah yang digunakan jika pengguna ingin mengubah data kelompok makanan. Lalu tombol keempat adalah tombol

hapus yang digunakan jika pengguna yang ingin menghapus data kelompok makanan. Apabila pengguna menekan tombol tambah dan ubah maka akan menuju ke antarmuka selanjutnya untuk menambahkan dan mengubah data kelompok makanan.

4.1.1.5. Antarmuka Tambah Kelompok Makanan

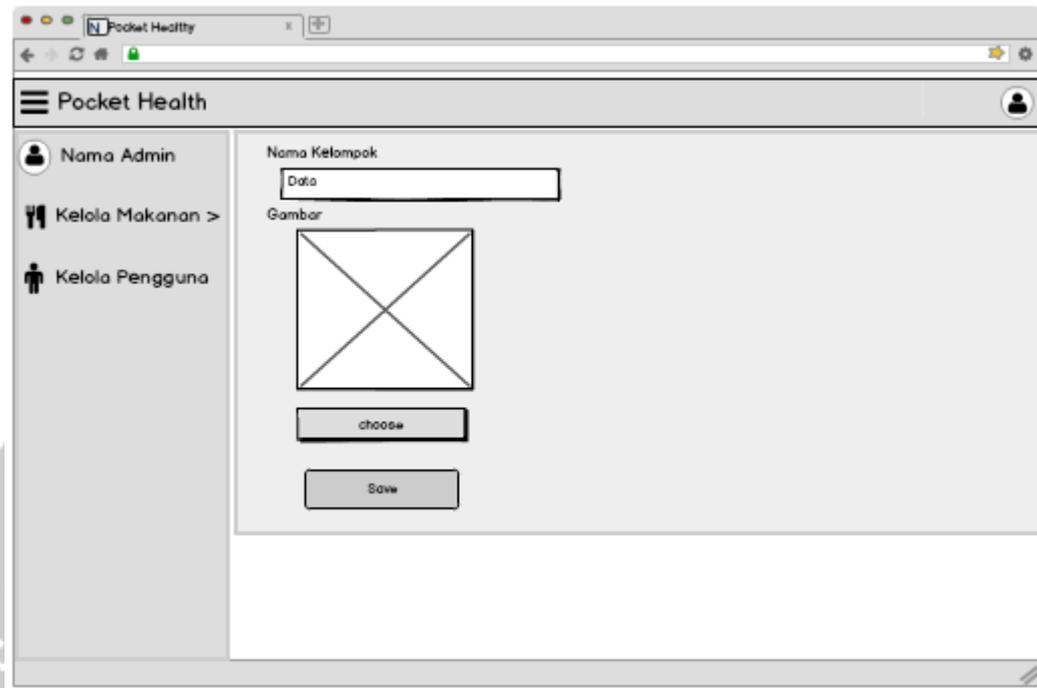


The screenshot shows a web browser window titled "Pocket Health". On the left is a sidebar menu with three items: "Nama Admin" (with a person icon), "Kelola Makanan >" (with a fork and knife icon), and "Kelola Pengguna" (with a person icon). The main content area is titled "Antarmuka Tambah Kelompok Makanan". It contains a form with the following elements: a text input field labeled "Nama Kelompok", a section labeled "Gambar" containing a "choose" button, and a "Save" button at the bottom.

Gambar 4. 5 Antarmuka Tambah Kelompok Makanan

Antarmuka gambar 4.5 digunakan untuk menambahkan data kelompok makanan. Pengguna harus memasukkan nama kelompok makanan dan memilih gambar kelompok makanan terlebih dahulu pada textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data kelompok makanan akan disimpan di database.

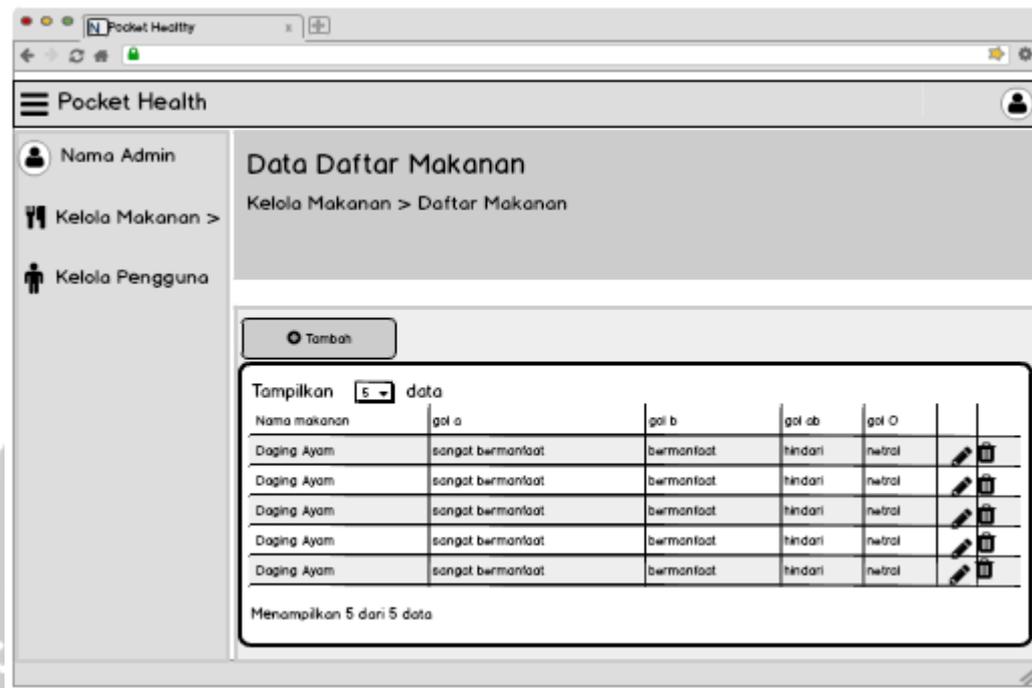
4.1.1.6. Antarmuka Ubah Kelompok Makanan



Gambar 4. 6 Antarmuka Ubah Kelompok Makanan

Antarmuka gambar 4.6 digunakan untuk mengubah data kelompok makanan yang sudah tersimpan pada database. Pengguna harus memasukkan data yang akan diubah yaitu nama kelompok makanan atau gambar kelompok makanan pada textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data kelompok makanan yang baru akan disimpan di database.

4.1.1.7. Antarmuka Kelola Daftar Makanan

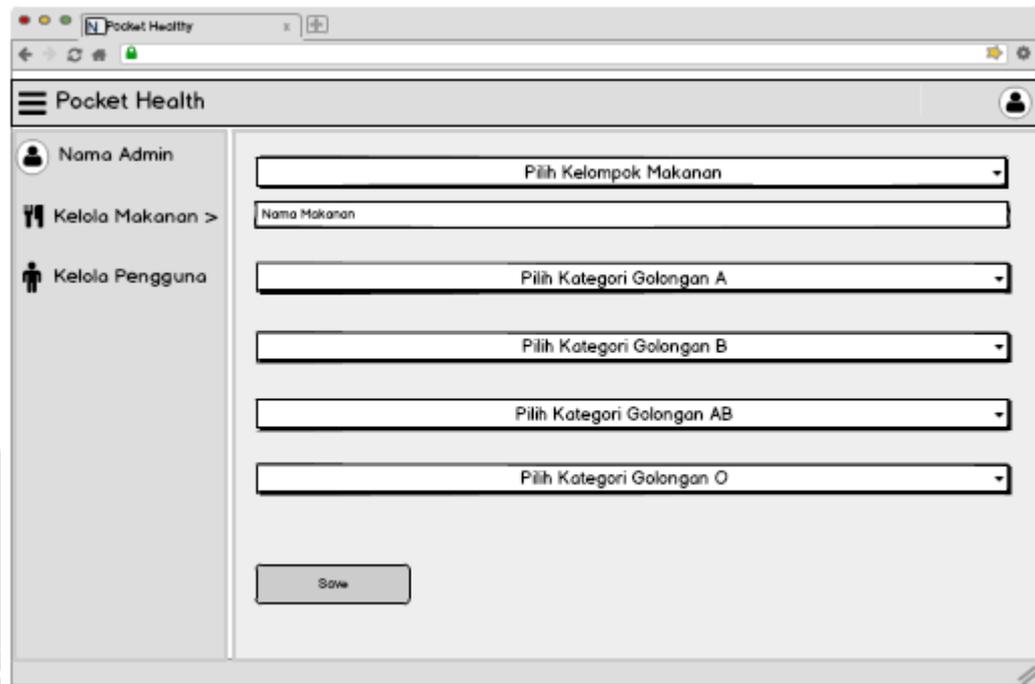


Gambar 4. 7 Antarmuka Kelola Daftar Makanan

Antarmuka gambar 4.7 digunakan untuk mengelola data daftar makanan. Pada antarmuka ini menampilkan data daftar. Data ditampilkan perhalaman yang dapat diatur melalui dropdown untuk menampilkan data, dan terdapat kolom pencarian untuk mencari data tertentu. Data yang tampil juga dapat di urutkan berdasarkan huruf dari kecil ke besar atau sebaliknya.

Ada tiga tombol dalam antarmuka ini, yang pertama adalah tombol tambah yang digunakan jika pengguna ingin menambahkan data daftar makanan. Lalu tombol ketiga adalah tombol ubah yang digunakan jika pengguna yang ingin mengubah data daftar makanan. Apabila pengguna menekan tombol tambah dan ubah maka akan menuju ke antarmuka selanjutnya untuk menambahkan dan mengubah data daftar makanan.

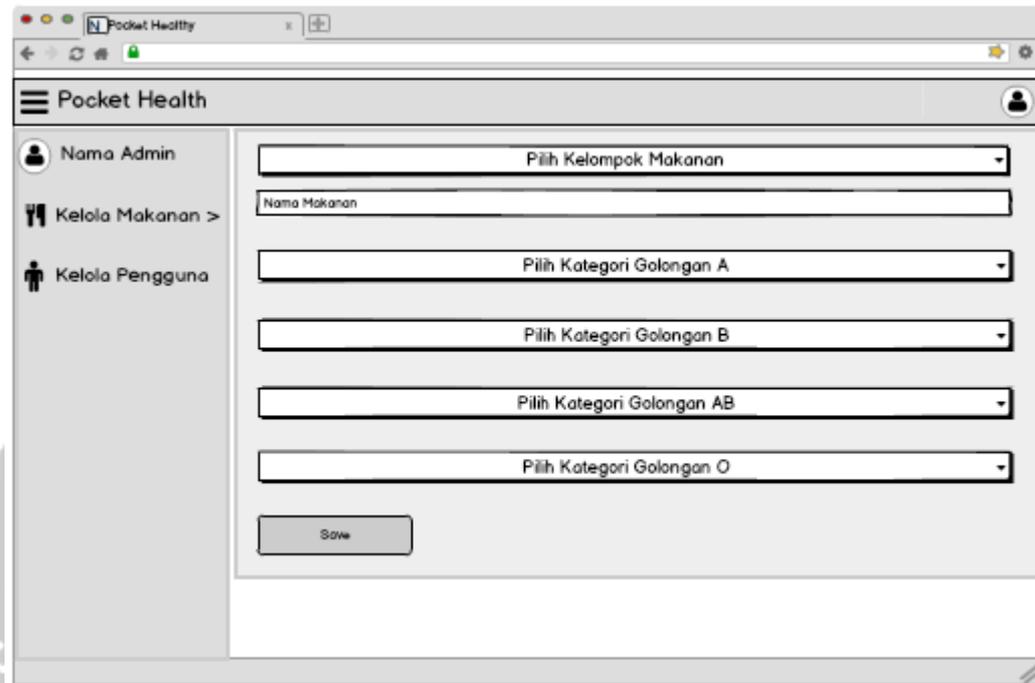
4.1.1.8. Antarmuka Tambah Daftar Makanan



Gambar 4. 8 Antarmuka Tambah Daftar Makanan

Antarmuka gambar 4.12 digunakan untuk menambahkan data daftar makanan. Pengguna harus memasukkan data daftar makanan terlebih dahulu pada dropdown dan textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data daftar makanan akan disimpan di database.

4.1.1.9. Antarmuka Ubah Daftar Makanan

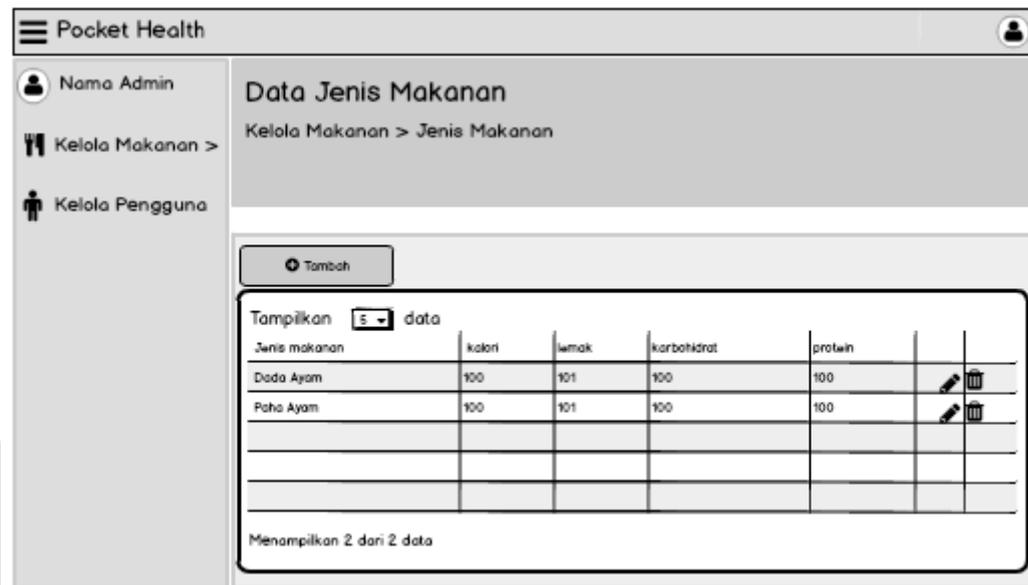


The screenshot shows a web browser window titled "Pocket Health". The interface has a sidebar on the left with a hamburger menu icon and a user profile icon. The sidebar contains the following items: "Nama Admin" with a user icon, "Kelola Makanan >" with a fork and knife icon, and "Kelola Pengguna" with a person icon. The main content area contains a form with the following elements: a dropdown menu labeled "Pilih Kelompok Makanan", a text input field labeled "Nama Makanan", a dropdown menu labeled "Pilih Kategori Golongan A", a dropdown menu labeled "Pilih Kategori Golongan B", a dropdown menu labeled "Pilih Kategori Golongan AB", a dropdown menu labeled "Pilih Kategori Golongan O", and a "Save" button at the bottom.

Gambar 4. 9 Antarmuka Ubah Daftar Makanan

Antarmuka gambar 4.9 digunakan untuk mengubah data daftar makanan yang sudah tersimpan pada database. Pengguna harus memasukkan data yang akan pada dropdown dan textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data daftar makanan yang baru akan disimpan di database.

4.1.1.10. Antarmuka Kelola Jenis Makanan

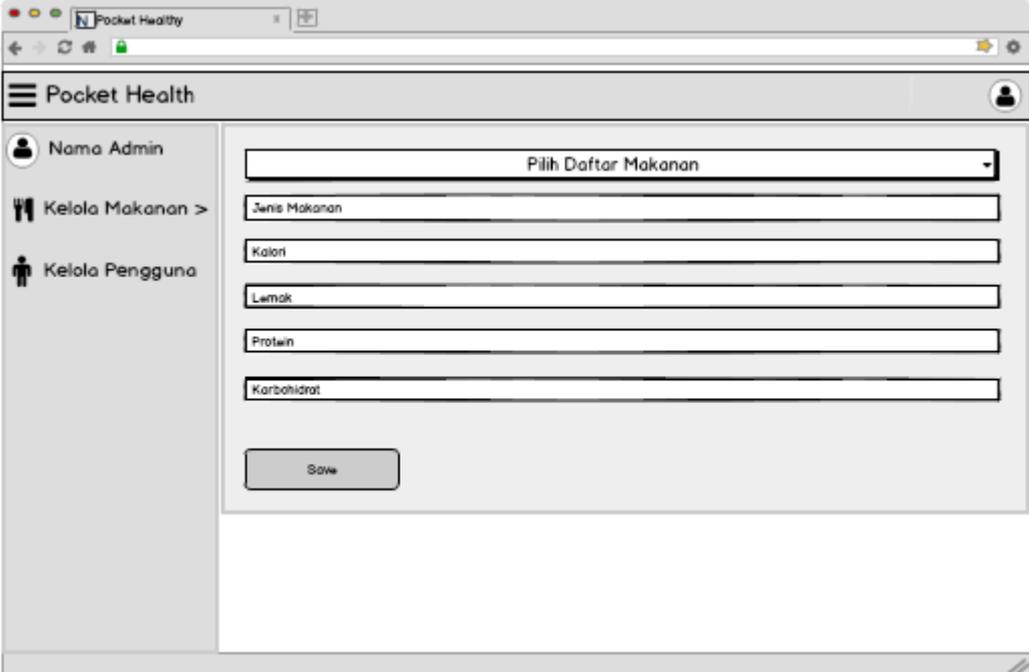


Gambar 4. 10 Antarmuka Kelola Jenis Makanan

Antarmuka gambar 4.10 digunakan untuk mengelola data jenis makanan. Pada antarmuka ini menampilkan data jenis. Data ditampilkan perhalaman yang dapat diatur melalui dropdown untuk menampilkan data, dan terdapat kolom pencarian untuk mencari data tertentu. Data yang tampil juga dapat di urutkan berdasarkan huruf dari kecil ke besar atau sebaliknya.

Ada tiga tombol dalam antarmuka ini, yang pertama adalah tombol tambah yang digunakan jika pengguna ingin menambahkan data jenis makanan. Lalu tombol ketiga adalah tombol ubah yang digunakan jika pengguna yang ingin mengubah data jenis makanan. Apabila pengguna menekan tombol tambah dan ubah maka akan menuju ke antarmuka selanjutnya untuk menambahkan dan mengubah data jenis makanan.

4.1.1.11. Antarmuka Tambah Jenis Makanan

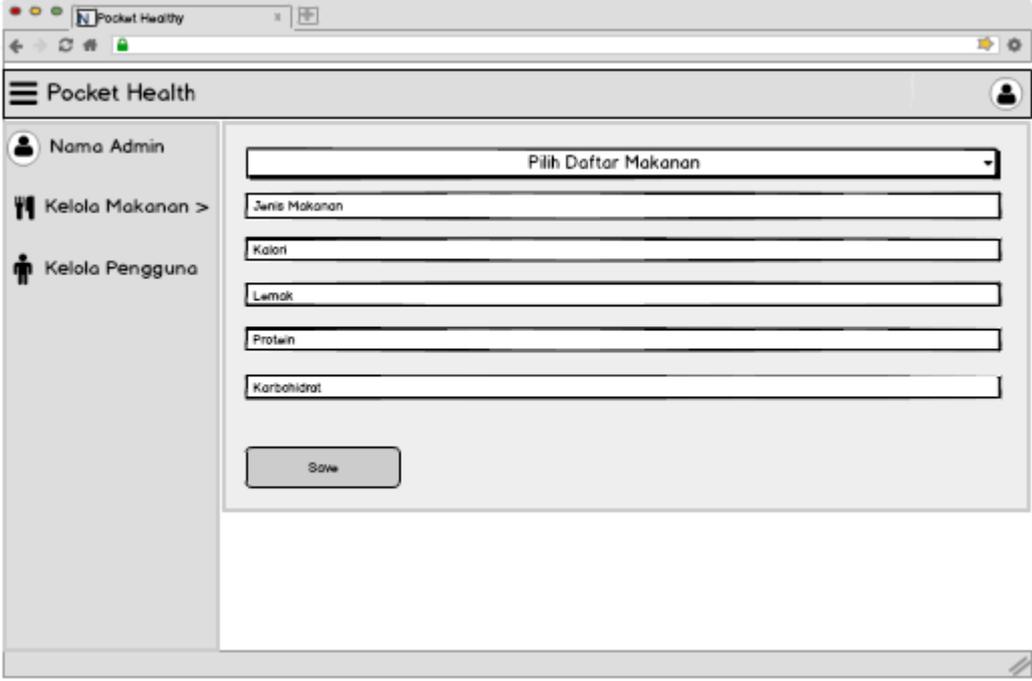


The screenshot shows a web browser window titled 'Pocket Health'. On the left is a sidebar with a hamburger menu icon and three items: 'Nama Admin' with a person icon, 'Kelola Makanan >' with a fork and knife icon, and 'Kelola Pengguna' with a person icon. The main content area has a header 'Pocket Health' with a user profile icon on the right. Below the header is a dropdown menu labeled 'Pilih Daftar Makanan'. Underneath are five text input fields labeled 'Jenis Makanan', 'Kalori', 'Lemak', 'Protein', and 'Karbohidrat'. At the bottom of the form is a 'Save' button.

Gambar 4. 11 Antarmuka Tambah Jenis Makanan

Antarmuka gambar 4.11 digunakan untuk menambahkan data jenis makanan. Pengguna harus memasukkan data jenis makanan terlebih dahulu pada dropdown dan textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data jenis makanan akan disimpan di database.

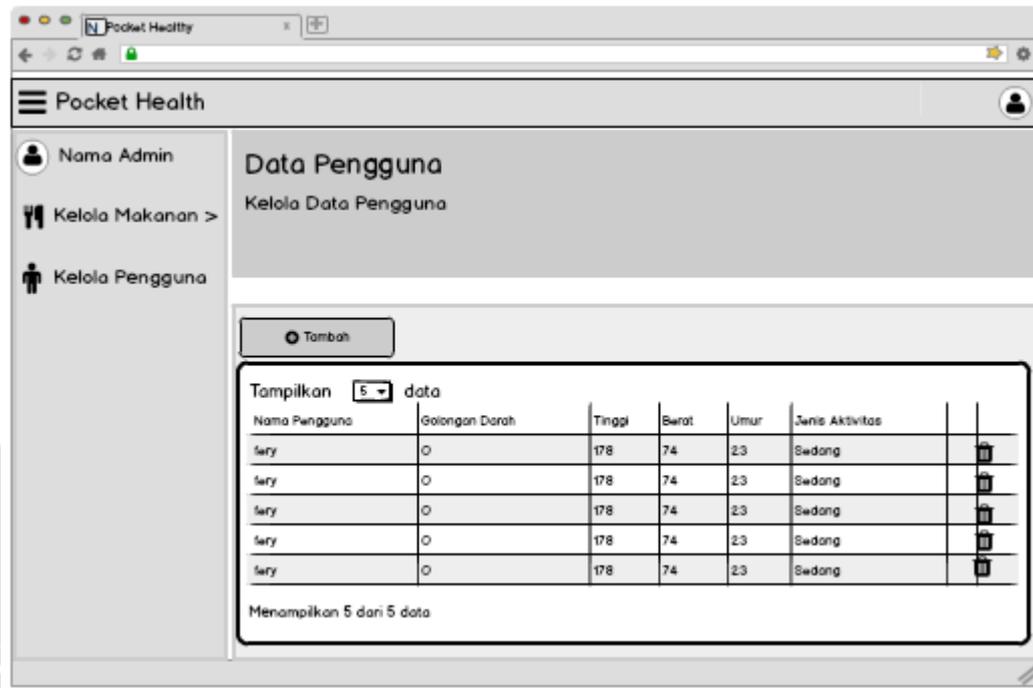
4.1.1.12. Antarmuka Ubah Jenis Makanan



Gambar 4. 12 Antarmuka Ubah Jenis Makanan

Antarmuka gambar 4.12 digunakan untuk mengubah data jenis makanan yang sudah tersimpan pada database. Pengguna harus memasukkan data yang akan pada dropdown dan textbox yang telah disediakan. Pada saat tombol simpan ditekan, sistem akan mengecek apakah data yang telah dimasukan sudah benar. Jika sudah maka data jenis makanan yang baru akan disimpan di database.

4.1.1.13. Antarmuka Kelola Pengguna



Gambar 4. 13 Antarmuka Kelola Pengguna

Antarmuka gambar 4.13 digunakan untuk mengelola data pengguna. Pada antarmuka ini menampilkan data pengguna. Data ditampilkan perhalaman yang dapat diatur melalui dropdown untuk menampilkan data, dan terdapat kolom pencarian untuk mencari data tertentu. Data yang tampil juga dapat di urutkan berdasarkan huruf dari kecil ke besar atau sebaliknya.

Ada satu tombol dalam antarmuka ini, tombol hapus yang digunakan untuk menghapus data pengguna tertentu yang terdaftar di database.

4.1.2. Sketsa UI Perangkat Lunak Mobile PH dan Deskripsinya

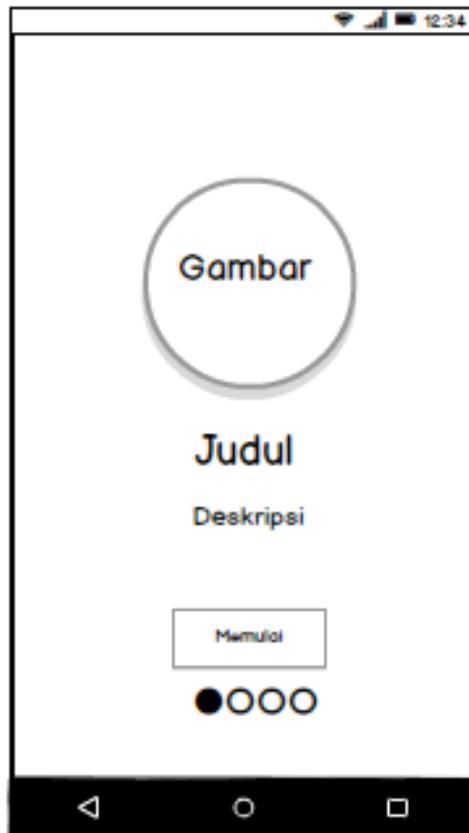
4.1.2.1. Antarmuka Halaman Splash



Gambar 4. 14 Antarmuka Halaman Splash

Antarmuka gambar 4.14 merupakan splash screen sebelum masuk keaplikasi yang berisi logo aplikasi. Antarmuka ini dijumpai pengguna setiap membuka aplikasi TRIPP dari keadaan tertutup.

4.1.2.2. Antarmuka Halaman Intro



Gambar 4. 15 Antarmuka Halaman Intro

Gambar 4.15 merupakan antarmuka intro aplikasi yang dijumpai pengguna pertama kali menginstal aplikasi atau pertama kali menggunakan aplikasi. Terdapat sebuah tombol memulai. Ketika tombol memulai di tekan maka akan berlanjut ke antarmuka login.

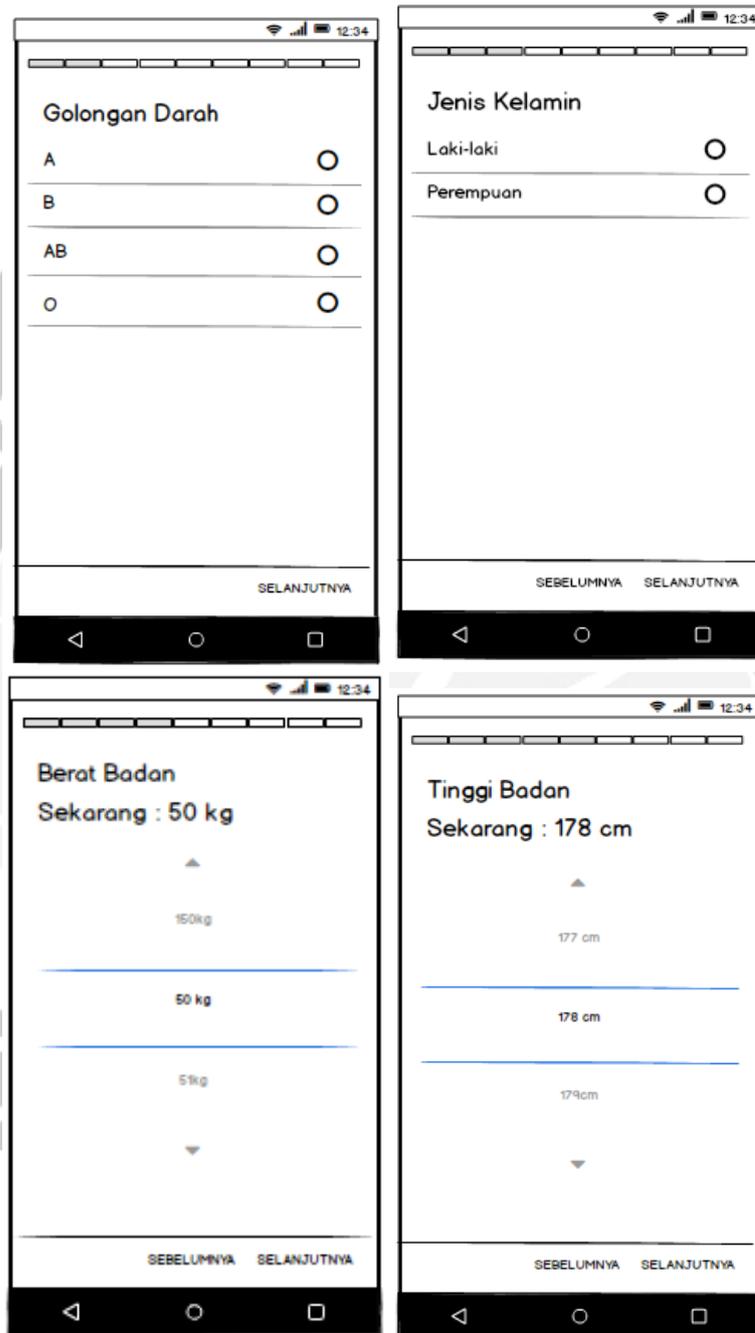
4.1.2.3. Antarmuka Halaman Login



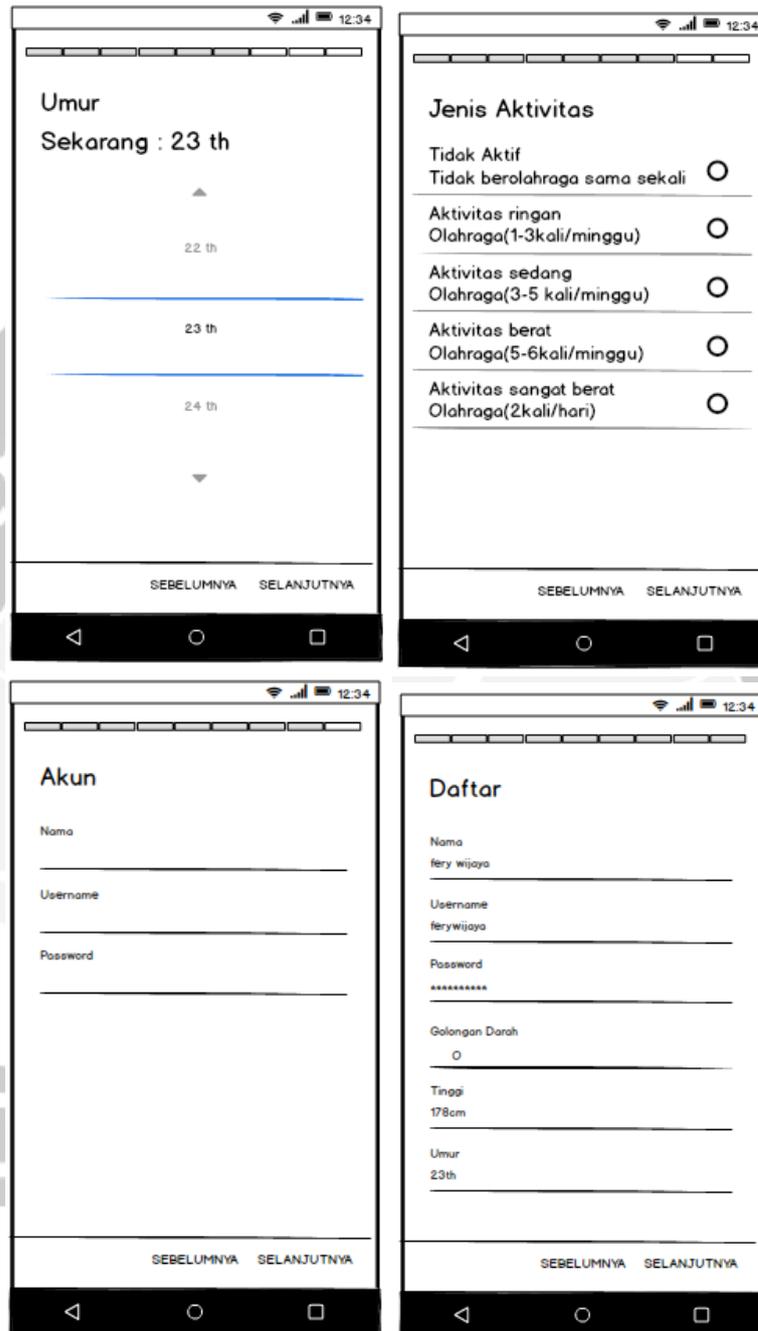
Gambar 4. 16 Antarmuka Halaman Login

Antarmuka gambar 4.16 digunakan untuk masuk kedalam aplikasi. Untuk masuk ke dalam aplikasi dibutuhkan data username dan password, maka disini terdapat dua edittext yaitu untuk username dan password. Lalu terdapat satu tombol untuk masuk. Jika tombol ditekan maka pengguna akan masuk ke dalam aplikasi. Kemudian terdapat juga textview untuk mendaftar jika belum memiliki akun untuk login. Jika textview daftar di tekan maka pengguna akan masuk untuk mendaftarkan sebuah akun.

4.1.2.4. Antarmuka Halaman Mendaftar



Gambar 4. 17 Antarmuka Halaman Mendaftar



Gambar 4. 18 Antarmuka Halaman Mendaftar

Antarmuka gambar 4.17 dan 4.18 digunakan untuk mendaftarkan sebuah akun pengguna dimana terdapat 7 langkah yang harus dilengkapi terlebih dahulu untuk membuat sebuah akun. Antarmuka ini akan dijumpai

pengguna ketika pertama kali menggunakan aplikasi dan diharuskan untuk mendaftar terlebih dahulu.

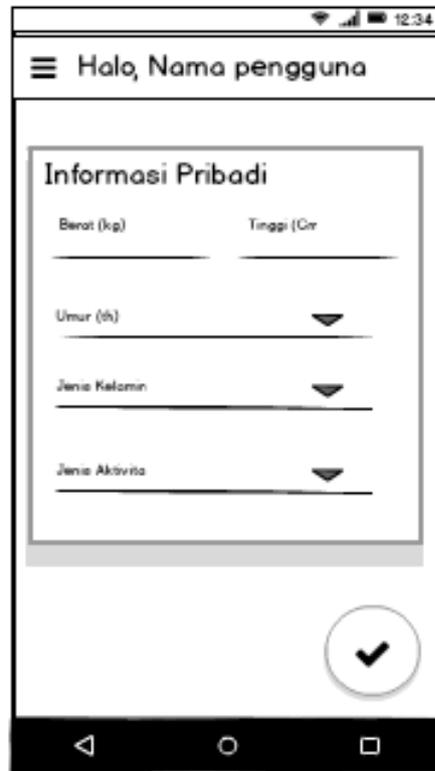
4.1.2.5. Antarmuka Halaman Beranda



Gambar 4. 19 Antarmuka Halaman Beranda

Antarmuka gambar 4.19 merupakan antarmuka awal ketika pengguna sudah melakukan login. Digunakan untuk menampilkan informasi pengguna yang berkaitan dengan informasi berat badan dan juga kebutuhan kalori pengguna serta terdapat juga button untuk menghitung ulang kalori dan informasi indeks massa tubuh jika terjadi perubahan berat.

4.1.2.6. Antarmuka Update Informasi Pribadi

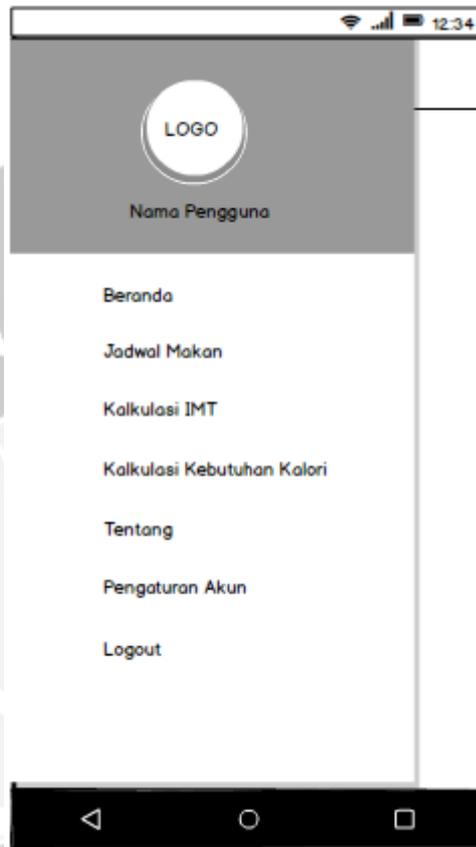


The screenshot shows a mobile application interface for updating personal information. At the top, there is a status bar with the time 12:34 and signal strength indicators. Below the status bar, there is a header with a hamburger menu icon and the text "Halo, Nama pengguna". The main content area is titled "Informasi Pribadi" and contains several input fields: "Berat (kg)", "Tinggi (Cm)", "Umur (th)", "Jenis Kelamin", and "Jenis Aktivitas". Each field has a dropdown arrow on the right side. At the bottom right of the form, there is a circular button with a checkmark icon. The background of the page features a large watermark logo with the text "serviens in veritatis".

Gambar 4. 20 Antarmuka Update Informasi Pribadi

Gambar 4.20 merupakan antarmuka update informasi pengguna yang digunakan untuk menghitung ulang informasi yang sudah ditampilkan pada antarmuka beranda.

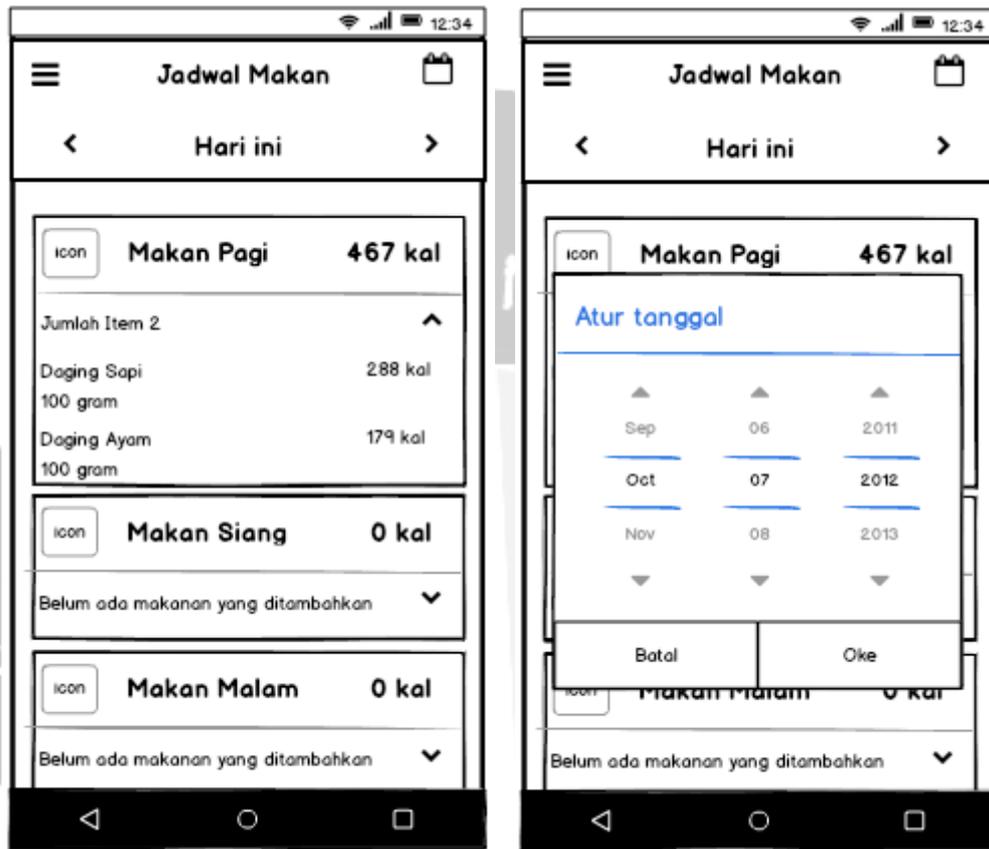
4.1.2.7. Antarmuka Navigation Drawer / Menu



Gambar 4. 21 Antarmuka Navigation Drawer

Antarmuka gambar 4.21 merupakan antarmuka menu ketika pengguna sudah melakukan login. Digunakan untuk menampilkan menu dan fitur yang dimiliki oleh aplikasi.

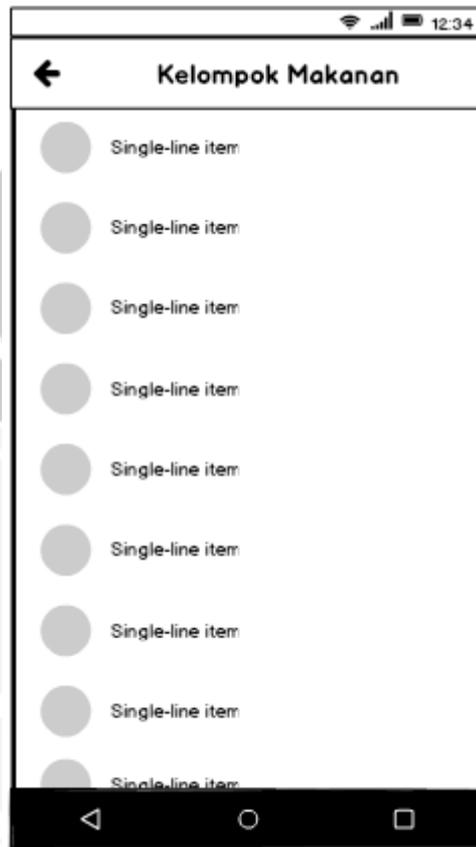
4.1.2.8. Antarmuka Halaman Jadwal Makan



Gambar 4. 22 Antarmuka Jadwal Makan

Antarmuka gambar 4.22 merupakan antarmuka ketika pengguna memilih menu jadwal makan. Digunakan untuk membantu pengguna menyusun jadwal makan sesuai golongan darah dan jumlah kalori. Pengaturan jadwal makan dilakukan dengan cara pengguna memilih tanggal mana saja yang ingin diatur dan kemudian menekan tombol tambah untuk menambahkan jadwal, ketika tombol tambah ditekan maka akan berlanjut ke antarmuka kelompok makanan.

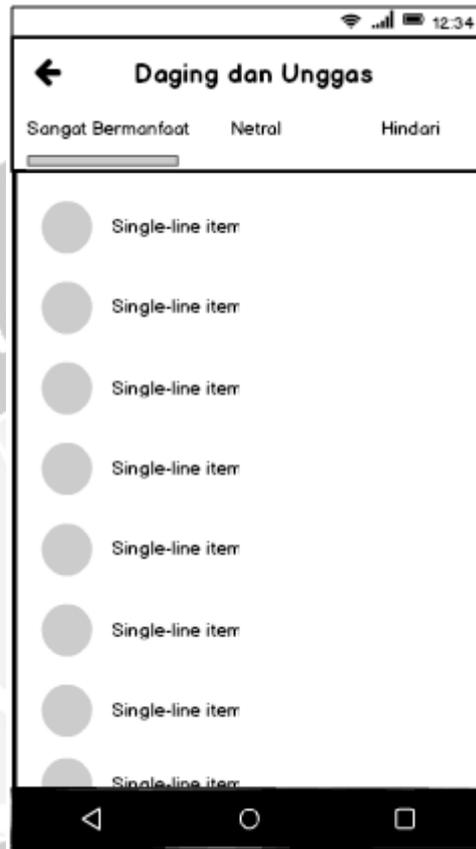
4.1.2.9. Antarmuka Halaman Kelompok Makanan



Gambar 4. 23 Antarmuka Kelompok Makanan

Antarmuka gambar 4.23 digunakan untuk menampilkan kelompok makanan yang di kelompokkan berdasarkan kelompok makanan berdasarkan golongan darah dan ketika list ditekan maka akan menampilkan daftar makan sesuai dengan kelompok makanan berdasarkan golongan darah.

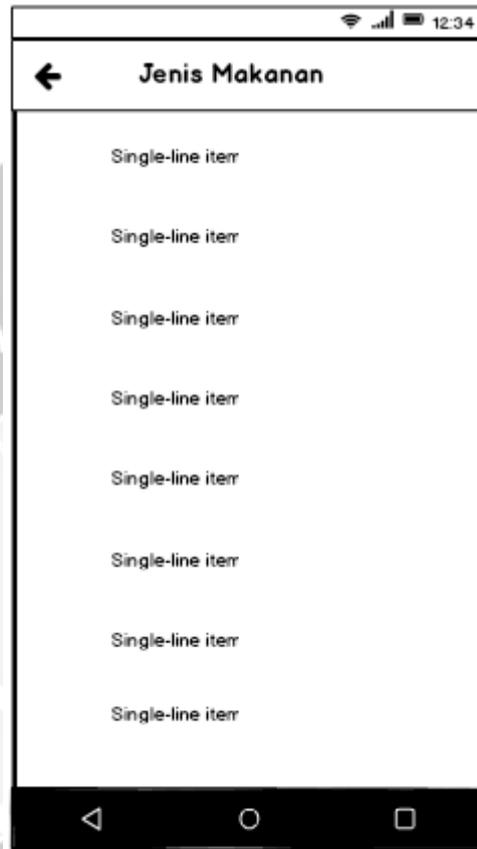
4.1.2.10. Antarmuka Halaman Daftar Makanan



Gambar 4. 24 Antarmuka Halaman Daftar Makanan

Antarmuka gambar 4.24 digunakan untuk menampilkan daftar makanan yang di kelompokkan berdasarkan kelompok makanan berdasarkan golongan darah dan ketika list ditekan maka akan menampilkan jenis makanan sesuai dengan daftar makanan berdasarkan golongan darah.

4.1.2.11. Antarmuka Halaman Jenis Makanan



Gambar 4. 25 Antarmuka Halaman Jenis Makanan

Antarmuka gambar 4.25 digunakan untuk menampilkan jenis makanan yang di kelompokkan berdasarkan daftar makanan berdasarkan golongan darah dan ketika list ditekan maka akan menampilkan dialog detail jenis makanan.

4.1.2.12. Antarmuka Hitung Kebutuhan Kalori Harian

The screenshot shows a mobile application interface titled "Kalkulasi Kebutuhan Kalori". At the top, there is a status bar with signal strength, Wi-Fi, and battery icons, and the time 12:34. Below the title bar, there are five input fields: "Berat (kg)", "Tinggi (Cm)", "Umur (th)", "Jenis Kelamin", and "Jenis Aktivitas". Each field has a horizontal line for text entry and a downward-pointing triangle indicating a dropdown menu. Below the input fields, there is a large rectangular box with the word "HASIL" in the center. To the right of the "HASIL" box is a circular button containing a calculator icon. At the bottom of the screen, there is a standard Android navigation bar with back, home, and recent apps buttons.

Gambar 4. 26 Antarmuka Hitung Kebutuhan Kalori Harian

Gambar 4.26 merupakan antarmuka hitung kebutuhan kalori. Antarmuka ini digunakan oleh pengguna jika ingin melakukan kalkulasi kebutuhan kalori saja. Pengguna bisa menekan tombol untuk menghitung ketika *semua data* sudah terisi. Sistem akan menampilkan hasil dari perhitungan.

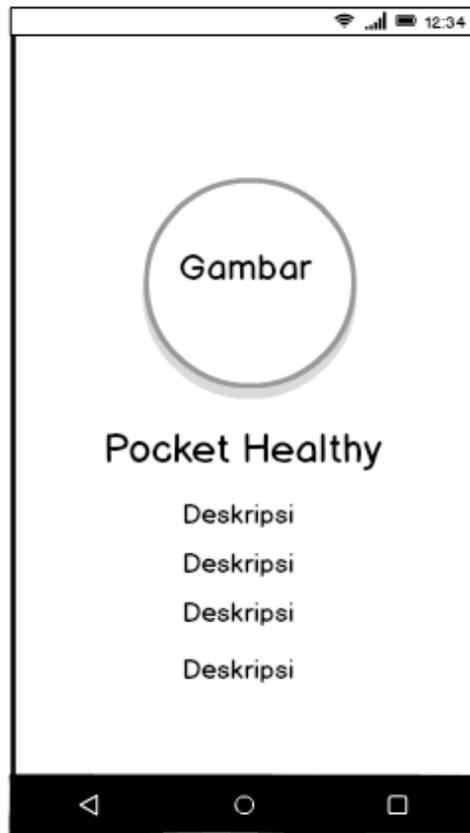
4.1.2.13. Antarmuka Hitung Indeks Massa Tubuh



Gambar 4. 27 Antarmuka Hitung Indeks Massa Tubuh

Gambar 4.27 merupakan antarmuka hitung indeks massa tubuh. Antarmuka ini digunakan oleh pengguna jika ingin melakukan kalkulasi indeks massa tubuh. Pengguna bisa menekan tombol untuk menghitung ketika *semua data* sudah terisi. Sistem akan menampilkan hasil dari perhitungan.

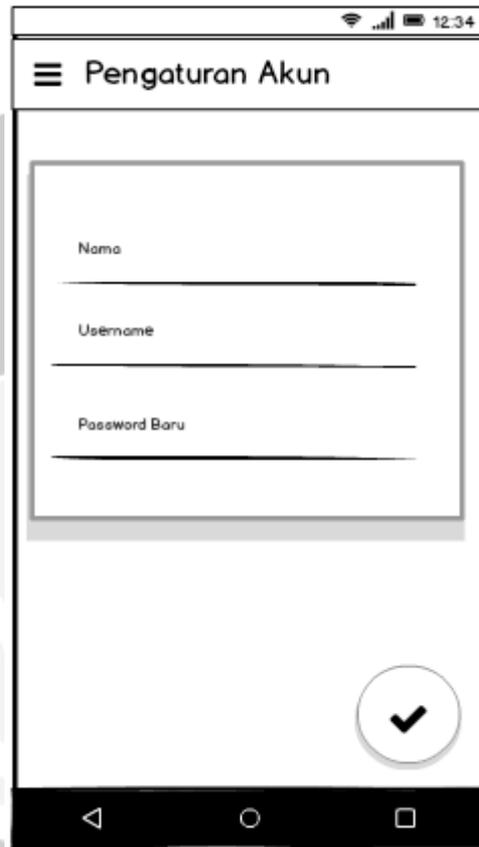
4.1.2.14. Antarmuka Tentang Aplikasi



Gambar 4. 28 Antarmuka Tentang Aplikasi

Gambar 4.28 merupakan antarmuka tentang aplikasi yang berisi keterangan mengenai pengetahuan singkat tentang aplikasi pocket healthy.

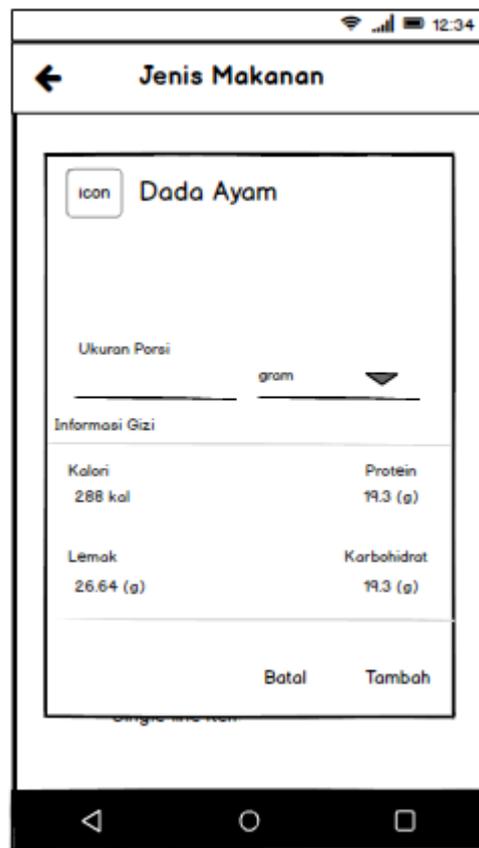
4.1.2.15. Antarmuka Pengaturan Akun



Gambar 4. 29 Antarmuka Pengaturan Akun

Gambar 4.29 merupakan antarmuka pengaturan akun untuk mengupdate data akun. Jika data masih kosong maka floating button dengan icon centang tidak dapat ditekan. Apabila data sudah di isi semua ketika ditekan maka akan menyimpan data baru ke *database*.

4.1.2.16. Antarmuka Detail Jenis Makanan



Gambar 4. 30 Antarmuka Dialog Tambah Makanan

Antarmuka Gambar 4.30 digunakan untuk menampilkan detail informasi dari jenis makanan yang ingin ditambahkan pada jadwal makanan. Pengguna memilih porsi makanan kemudian system akan menghitung dan menampilkan informasi kalori, lemak, karbohidrat, dan protein dari makanan yang dipilih. Terdapat tombol simpan yang digunakan untuk menyimpan makanan yang dipilih ke jadwal makan pengguna.

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

Pocket Healthy

(PH)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Fery Wijaya / 12 07 06802

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
	Fakultas Teknologi Industri	SKPL-PH		1/52
		<i>Revisi</i>		

DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

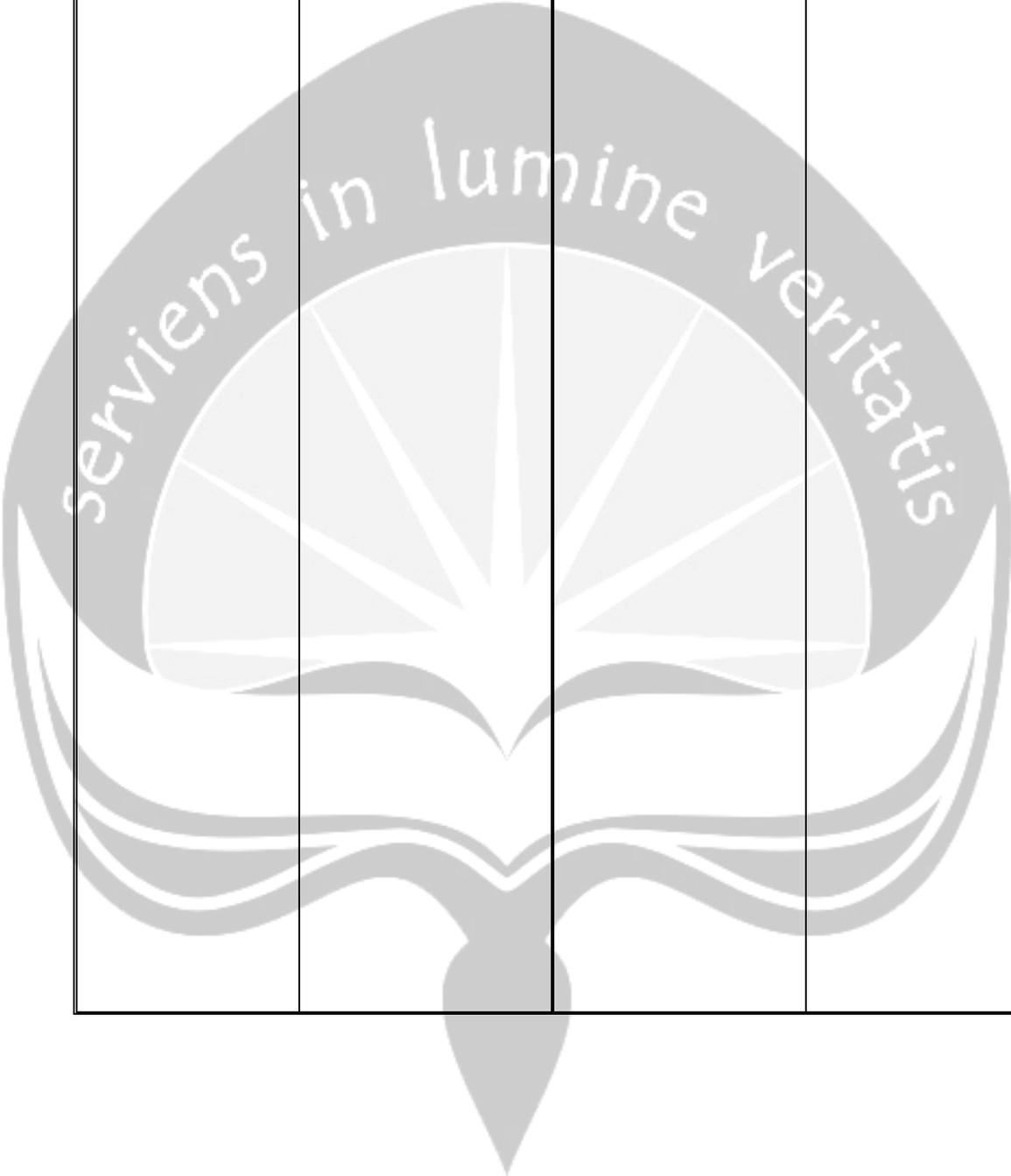
INDEX TGL	-	A	B	C	D	E	F
DITULIS OLEH	FW						
DIPERIKSA OLEH	YSP ERN						
DISETUJUI OLEH	YSP ERN						

Program Studi Teknik Informatika	SKPL – PH	2/ 52
--	-----------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi



Program Studi Teknik Informatika	SKPL – PH	3/ 52
--	-----------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

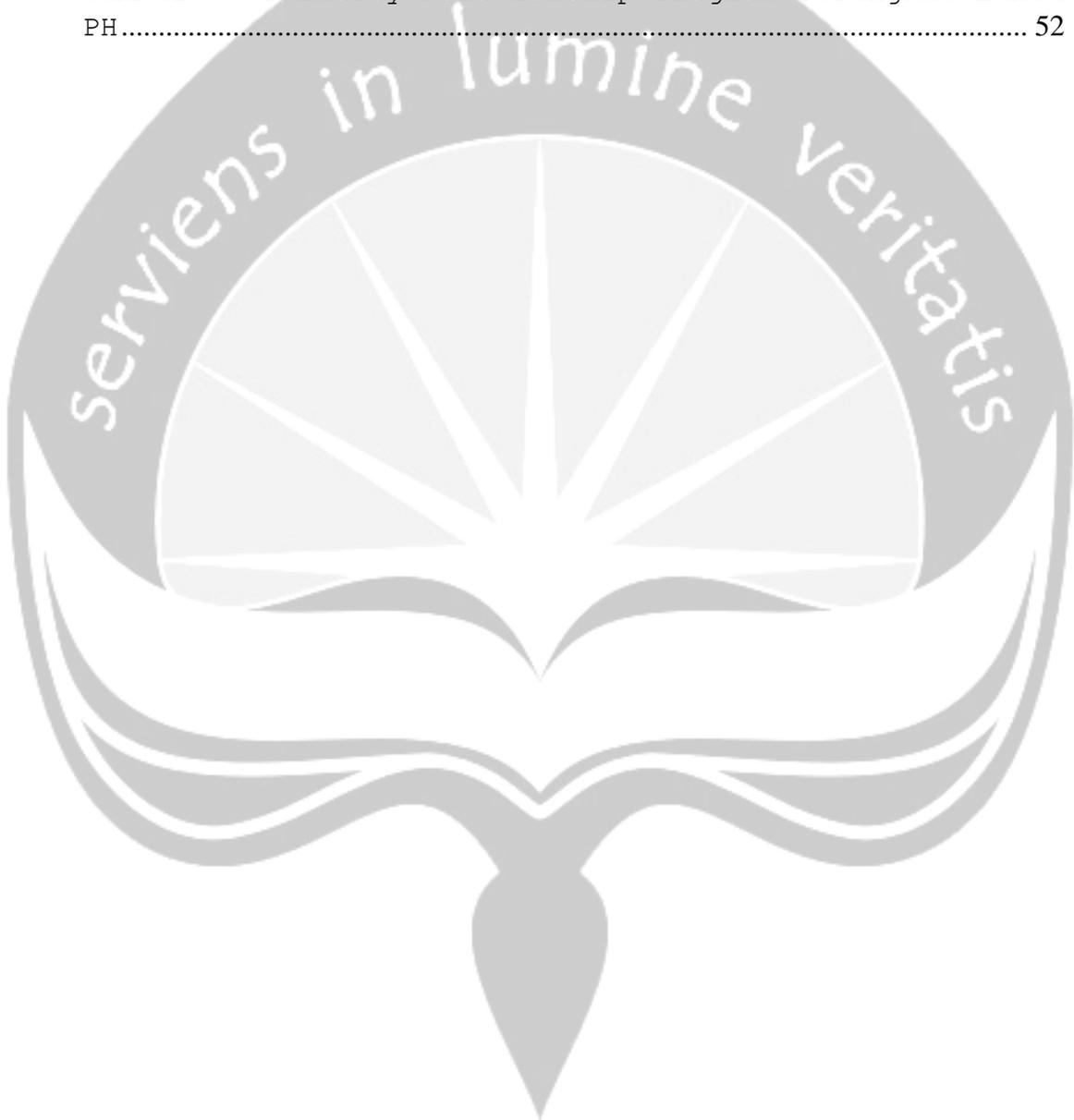
DAFTAR ISI

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim dan Singkatan	7
1.4	Referensi	9
1.5	Deskripsi umum (Overview)	9
2	Deskripsi Kebutuhan	10
2.1	Perspektif produk	10
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	18
2.4	Batasan-batasan	18
2.5	Asumsi dan Ketergantungan	18
3	Kebutuhan khusus	18
3.1	Kebutuhan antarmuka eksternal	18
3.2	Kebutuhan fungsionalitas Perangkat Lunak	21
4	Spesifikasi Rinci Kebutuhan	22
4.1	Spesifikasi Kebutuhan Fungsionalitas	22
5	ERD	52

Program Studi Teknik Informatika	SKPL – PH	4/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

DAFTAR GAMBAR

Gambar 2. 1	Arsitektur Perangkat Lunak PH.....	11
Gambar 3. 1	Use Case Diagram Perangkat Lunak Web.....	21
Gambar 3. 2	Use Case Diagram Perangkat Lunak Mobile....	22
Gambar 5. 1	Entity Relationship Diagram Perangkat Lunak PH.....	52



Program Studi Teknik Informatika	SKPL – PH	5/ 52
--	-----------	-------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak Pocket Healthy (PH) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna), dan atribut (*fitur-fitur* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-PH ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak PH dikembangkan dengan tujuan untuk:

1. Mengelola data makanan berdasarkan golongan darah dan melihat data pengguna berbasis web.
2. Menghitung dan menampilkan hasil dari indeks massa tubuh dan kebutuhan kalori harian berdasarkan masukan data berat, tinggi, umur, dan jenis aktivitas dari pengguna berbasis *mobile*.
3. Membantu mengatur jadwal makan harian dengan menu makanan berdasarkan golongan darah serta membatasi maksimal kalori harian sesuai data pengguna berbasis *mobile*.
4. Memudahkan pengguna dalam menjalankan diet sehat berdasarkan golongan darah.

Program Studi Teknik Informatika	SKPL – PH	6/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
PH	Perangkat lunak untuk memudahkan diet sehat berdasarkan golongan darah.
SKPL-PH-XXX	Kode yang merepresentasikan kebutuhan pada PH(Aplikasi mobile Pocket Healthy) dimana XXX merupakan nomor fungsi produk.
SKPL-PH-WEB-XXX	Kode yang merepresentasikan kebutuhan pada PH-WEB (Web untuk pengelolaan data untuk aplikasi mobile)dimana XXX merupakan nomor fungsi produk.
Web	Sejumlah halaman yang memiliki topik yang terkait, dapat juga disertai dengan gambar, video atau jenis lainnya. Biasanya dapat diakses menggunakan jaringan internet ataupun jaringan wilayah lokal (LAN)
Android	Sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layarsentuhseperti telepon pintar dan komputer tablet.

Program Studi Teknik Informatika	SKPL - PH	7/52
--	-----------	------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
PHP	Hypertext Preprocessor merupakan Bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Java	JAVA merupakan sebuah Bahasa pemrograman yang berorientasi OBJEK dan dapat dijalankan (<i>Run</i>) pada segala jenis sistem operasi (OS) karena perkembangan JAVA ini tidak hanya terfokus pada satu sistem operasi saja tetapi dikembangkan untuk segala macam jenis sistem operasi dan bersifat <i>open source</i> .
XML	Bahasa markup untuk keperluan umum yang disarankan oleh W3C untuk membuat dokumen markup keperluan pertukaran data antar sistem yang beraneka ragam.

Program Studi Teknik Informatika	SKPL – PH	8/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Fery Wijaya, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) MRS*, Universitas Atma Jaya Yogyakarta.
2. Proposal Tugas Akhir Pembangunan Aplikasi Diet Sehat Golongan Darah Berbasis Mobile.

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak PH yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak PH tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak PH yang akan dikembangkan.

Program Studi Teknik Informatika	SKPL – PH	9/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

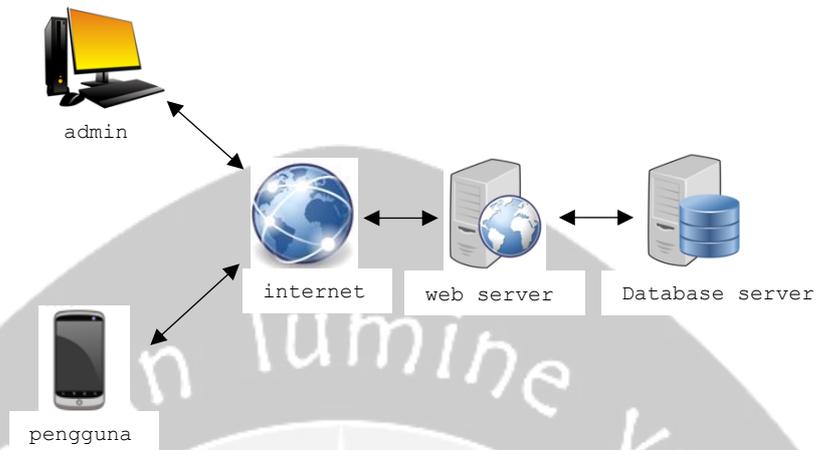
2 Deskripsi Kebutuhan

2.1 Perspektif produk

Pocket Healthy (PH) merupakan perangkat lunak yang dikembangkan dengan tujuan untuk memudahkan pengguna dalam menjalani diet sehat berdasarkan golongan darah, memperoleh informasi indeks masa tubuh untuk mengetahui apakah berat badan saat ini sudah ideal atau belum dan membantu mendapatkan informasi jumlah maksimal kebutuhan kalori harian serta pengaturan jadwal makan. PH diharapkan dapat menjadi sarana sumber informasi dan pengaturan jadwal makanan berdasarkan golongan darah yang dilengkapi dengan informasi gizi sehingga setiap makanan yang dikonsumsi sehingga dapat memenuhi kebutuhan pengguna.

Perangkat lunak PH dikembangkan pada perangkat mobile dan berjalan pada sistem operasi android. Sistem untuk mengelola data makanan berjalan pada web yang dapat dibuka di browser apapun. Sistem PH dibuat menggunakan bahasa pemrograman Java, XML dalam pembuatan layout. Perangkat lunak PH menggunakan web service sebagai jembatan komunikasi dari basis data ke perangkat lunak PH, dan untuk pengelolaan data makanan menggunakan PHP dengan framework Laravel. Lingkungan pemrogramannya menggunakan Android Studio dan Sublime Text 3. Pada database sistem menggunakan MySQL.

Program Studi Teknik Informatika	SKPL – PH	10/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



Gambar 2. 1 Arsitektur Perangkat Lunak PH
 Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (*Graphical User Interface*). Seperti yang terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa client server, dimana semua data disimpan di server. Pengguna dapat mengakses data yang ada di server tersebut secara online dengan memanggil web service pada website yang tersedia di web server.

Masukan data dari pengguna akan disimpan dalam database server, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke database server yang selanjutnya dikirimkan ke client yang meminta melalui web server.

2.2 Fungsi Produk

Fungsi produk perangkat lunak PH adalah sebagai berikut :

Program Studi Teknik Informatika	SKPL – PH	11/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

PERANGKAT LUNAK WEB

1. Fungsi *Login* (**SKPL-PH-Web-001**).

Fungsi yang digunakan oleh Administrator untuk dapat masuk dalam sistem yang akan digunakan.

2. Fungsi *Logout* (**SKPL-PH-Web-002**).

Fungsi yang digunakan oleh Administrator untuk keluar dari sistem yang digunakan.

3. Fungsi *Mengedit Profil* (**SKPL-PH-Web-003**).

Fungsi yang digunakan oleh Administrator untuk mengedit profil seperti nama, email, dan password serta avatar.

4. Fungsi *Mengelola Data Kelompok Makanan* (**SKPL-PH-Web-004**).

Fungsi yang digunakan Administrator untuk mengelola data kelompok makanan.

Fungsi *Pengelolaan Data Kelompok Makanan* meliputi:

a. Fungsi *Tambah Data Kelompok Makanan* (**SKPL-PH-Web-004-01**).

Fungsi yang digunakan untuk menambahkan data kelompok makanan ke dalam basis data.

b. Fungsi *Ubah Data Kelompok Makanan* (**SKPL-PH-Web-004-02**).

Fungsi yang digunakan untuk mengubah data kelompok makanan tertentu.

c. Fungsi *Tampil Data Kelompok Makanan* (**SKPL-PH-Web-004-03**).

Program Studi Teknik Informatika	SKPL – PH	12/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi yang digunakan untuk menampilkan data kelompok makanan.

- d. Fungsi Hapus Data Kelompok Makanan (**SKPL-PH-Web-004-04**).

Fungsi yang digunakan untuk menghapus data kelompok makanan tertentu.

- e. Fungsi Cari Data Kelompok Makanan (**SKPL-PH-Web-004-05**).

Fungsi yang digunakan untuk mencari data kelompok makanan sesuai dengan *keyword*.

5. Fungsi Megeelola Data Daftar Makanan (**SKPL-PH-Web-005**).

Fungsi yang digunakan Administrator untuk mengelola data daftar makanan.

Fungsi *Pengelolaan Data Daftar Makanan* meliputi:

- a. Fungsi Tambah Data Daftar Makanan (**SKPL-PH-Web-005-01**).

Fungsi yang digunakan untuk menambahkan data daftar makanan ke dalam basis data.

- b. Fungsi Ubah Data Daftar Makanan (**SKPL-PH-Web-005-02**).

Fungsi yang digunakan untuk mengubah data daftar makanan tertentu.

- c. Fungsi Tampil Data Daftar Makanan (**SKPL-PH-Web-005-03**).

Program Studi Teknik Informatika	SKPL - PH	13/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi yang digunakan untuk menampilkan data daftar makanan.

- d. Fungsi Hapus Data *Daftar Makanan* (**SKPL-PH-Web-005-04**).

Fungsi yang digunakan untuk menghapus data daftar makanan tertentu.

- f. Fungsi *Cari Data Daftar Makanan* (**SKPL-PH-Web-005-05**).

Fungsi yang digunakan untuk mencari data daftar makanan sesuai dengan *keyword*.

6. Fungsi *Megelola Data Jenis Makanan* (**SKPL-PH-Web-006**).

Fungsi yang digunakan Administrator untuk mengelola data jenis makanan.

Fungsi *Pengelolaan Data Daftar Makanan* meliputi:

- a. Fungsi *Tambah Data Jenis Makanan* (**SKPL-PH-Web-006-01**).

Fungsi yang digunakan untuk menambahkan data jenis makanan ke dalam basis data.

- b. Fungsi *Ubah Data Jenis Makanan* (**SKPL-PH-Web-006-02**).

Fungsi yang digunakan untuk mengubah data jenis makanan tertentu.

- c. Fungsi *Tampil Data Jenis Makanan* (**SKPL-PH-Web-006-03**).

Program Studi Teknik Informatika	SKPL – PH	14/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi yang digunakan untuk menampilkan data jenis makanan.

- d. Fungsi Hapus Data *Jenis Makanan* (**SKPL-PH-Web-006-04**).

Fungsi yang digunakan untuk menghapus data jenis makanan tertentu.

- g. Fungsi *Cari Data Jenis Makanan* (**SKPL-PH-Web-006-05**).

Fungsi yang digunakan untuk mencari data jenis makanan sesuai dengan *keyword*.

7. Fungsi *Mengelola Data Pengguna Aplikasi Mobile* (**SKPL-PH-Web-007**).

Merupakan fungsi yang digunakan pengguna untuk mengelola data pengguna.

Fungsi *Pengelolaan Pengguna* meliputi :

- a. Fungsi *Tampil Data Pengguna* (**SKPL-PH-Web-007-01**).

Fungsi yang digunakan untuk menampilkan data pengguna.

- b. Fungsi *Cari Data Pengguna* (**SKPL-PH-Web-007-02**).

Fungsi yang digunakan untuk mencari data pengguna.

- c. Fungsi *Hapus Data Pengguna* (**SKPL-PH-Web-007-03**).

Fungsi yang digunakan untuk menghapus data pengguna.

Program Studi Teknik Informatika	SKPL – PH	15/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

PERANGKAT LUNAK MOBILE

1. Fungsi *Mendaftar* (**SKPL-PH-001**).

Fungsi yang digunakan oleh pengguna untuk mendaftar.

2. Fungsi *Login* (**SKPL-PH-002**).

Fungsi yang digunakan oleh pengguna untuk dapat masuk dalam sistem yang akan digunakan.

3. Fungsi *Logout* (**SKPL-PH-003**).

Fungsi yang digunakan oleh pengguna untuk keluar dari sistem.

4. Fungsi *Menampilkan Informasi Kebutuhan Kalori Harian dan Indeks Massa Tubuh* (**SKPL-PH-004**).

Fungsi yang digunakan oleh pengguna untuk menampilkan informasi kebutuhan kalori harian dan indeks massa tubuh dari pengguna.

5. Fungsi *Mengupdate Informasi Pengguna* (**SKPL-PH-005**).

Fungsi yang digunakan oleh pengguna untuk mengupdate informasi dari pengguna yang digunakan untuk menghitung ulang jumlah kebutuhan kalori dan indeks massa tubuh dari pengguna.

6. Fungsi *Mengatur Jadwal Makan Harian* (**SKPL-PH-006**).

Fungsi yang digunakan oleh pengguna untuk dapat mengatur jadwal makan harian yang sesuai dengan golongan darah dari pengguna.

Program Studi Teknik Informatika	SKPL – PH	16/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

7. Fungsi Menghitung *Indeks Massa Tubuh* (**SKPL-PH-007**).

Fungsi yang digunakan oleh pengguna untuk menghitung indeks massa tubuh pengguna.

8. Fungsi Menghitung *Kebutuhan Kalori Harian* (**SKPL-PH-008**).

Fungsi yang digunakan oleh pengguna untuk menghitung jumlah kalori yang dibutuhkan dalam sehari.

9. Fungsi *Mengubah Akun* (**SKPL-PH-009**).

Fungsi yang digunakan oleh pengguna untuk mengubah informasi akun seperti username, password, dan nama pengguna.

10. Fungsi Menampilkan *Makanan Berdasarkan Golongan Darah* (**SKPL-PH-010**).

Fungsi yang digunakan untuk menampilkan daftar makanan berdasarkan kelompok serta daftar makanan dan jenis makanan yang sesuai dengan golongan darah.

11. Fungsi Memberikan Notifikasi Jadwal Makan (**SKPL-PH-011**).

Fungsi yang digunakan untuk memberikan notifikasi pengingat waktu makan berdasarkan jadwal makan yang sudah diatur.

Program Studi Teknik Informatika	SKPL – PH	17/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak PH adalah sebagai berikut :

1. Memahami pengoperasian PC Desktop ataupun Laptop yang mendukung pengaksesan Web.
2. Memahami pengoperasian perangkat telepon pintar dengan sistem operasi Android.
3. Mengerti penggunaan perangkat lunak PH.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak PH tersebut adalah :

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak PH.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada web browser apapun, *smartphone* yang menggunakan sistem operasi android dan membutuhkan koneksi internet.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak PH meliputi kebutuhan antarmuka pemakai, antarmuka

Program Studi Teknik Informatika	SKPL – PH	18/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk halaman web untuk pengelolaan admin dan dalam bentuk aplikasi android untuk pengguna.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan untuk menjalankan perangkat lunak PH adalah:

1. Perangkat PC/Laptop
Spesifikasi minimum :
 - RAM : 512 MB
 - Hardisk : 320 GB
 - CPU : 1,2 GHz
 - Monitor, Keyboard, Mouse.
2. Perangkat Android Smartphome.
Spesifikasi minimum :
 - RAM : 512 MB
 - Hardisk : 1 GB
 - CPU : 1,2 GHz
3. Perangkat Database Server.
Spesifikasi minimum :
 - RAM : 32 GB
 - Hardisk : 300 GB
 - CPU : 2 Core
 - Ethernet LAN : 2 port
4. Perangkat Web Server.
Spesifikasi minimum :

Program Studi Teknik Informatika	SKPL – PH	19/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

- RAM : 16 GB
- Hardisk : 250 GB
- CPU : 2 Core
- Ethernet LAN : 2 port

3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak PH adalah sebagai berikut :

1. Nama : MySQL
 Sumber : Oracle.
 Sebagai database management system (DBMS) yang digunakan untuk penyimpanan data di sisi server.
2. Nama : Windows XP/Vista/7/8
 Sumber : Microsoft.
 Sebagai sistem operasi yang digunakan.
3. Nama : Apache
 Sumber : Apache Software Foundation.
 Sebagai web server.
4. Nama : PHP
 Sumber : PHP Group
 Sebagai bahasa pemrograman pada sisi server.
5. Nama : IE/Firefox/Chrome/dll
 Sumber : Microsoft/Mozilla/Google/dll
 Sebagai web browser untuk membuka sistem web.
6. Nama : Android OS
 Sumber : Google Android
 Sebagai aplikasi sistem operasi untuk menjalankan aplikasi Pocket Healthy dalam sistem mobile.

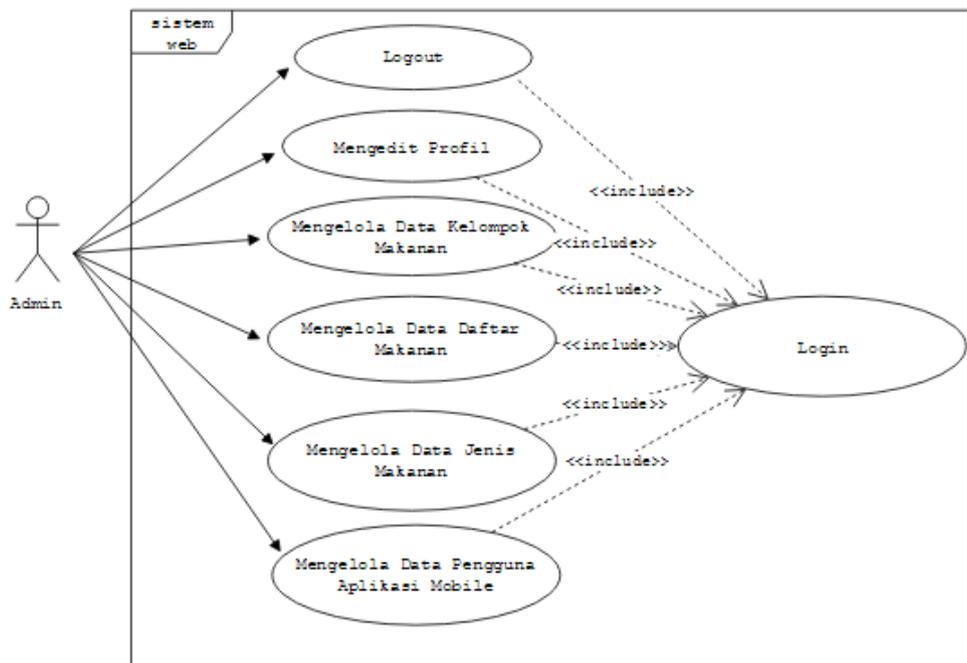
Program Studi Teknik Informatika	SKPL – PH	20/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak PH menggunakan protocol TCP/IP, sedangkan untuk aplikasi web menggunakan protocol HTTP.

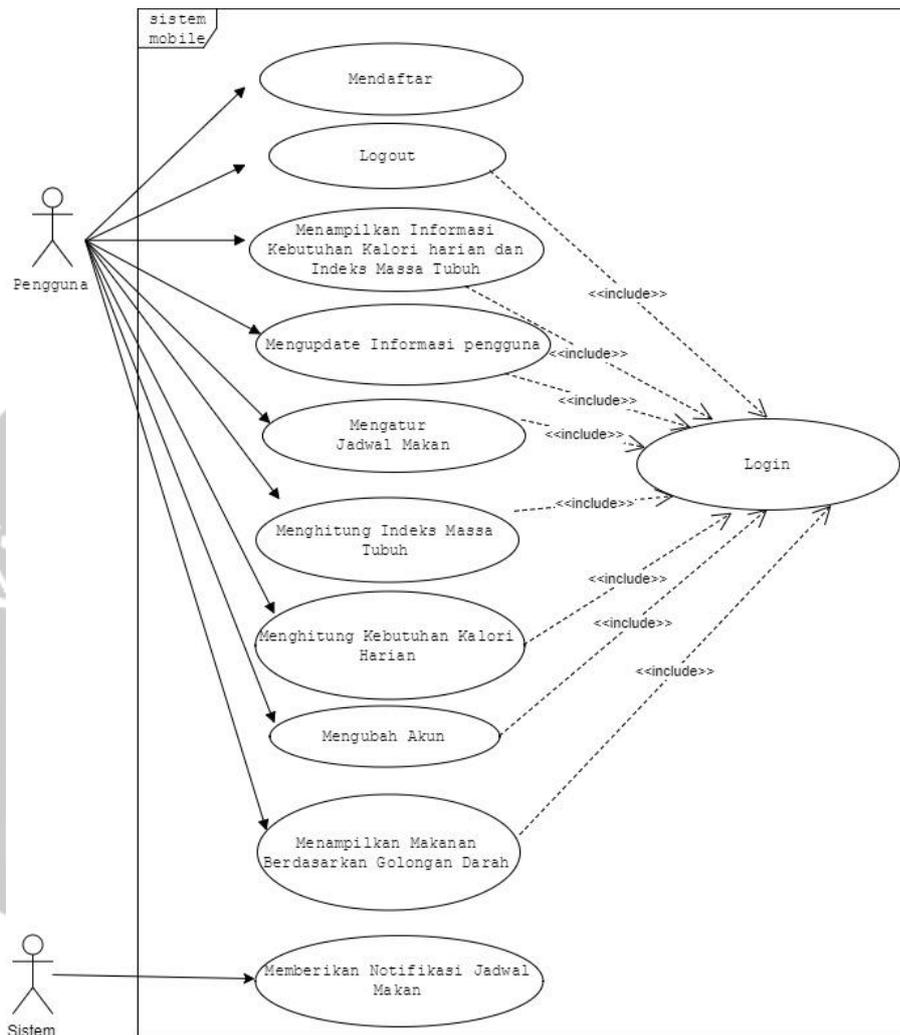
3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 3. 1 Use Case Diagram Perangkat Lunak Web

Program Studi Teknik Informatika	SKPL – PH	21/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		



Gambar 3. 2 Use Case Diagram Perangkat Lunak Mobile

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

PERANGKAT LUNAK WEB

4.1.1 Use case Spesification : Login

1. Brief Description

Use case ini digunakan untuk memperoleh akses masuk ke sistem yang digunakan oleh Administrator yang telah terdaftar di dalam basis data.

Program Studi Teknik Informatika	SKPL – PH	22/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login.
2. Sistem menampilkan antarmuka untuk login.
3. Aktor menginputkan email dan password.
4. Sistem melakukan validasi terhadap email dan password yang telah diinputkan.

E-1 email atau password yang diinputkan aktor salah.

5. Sistem memberikan akses masuk ke dalam sistem.
6. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 email atau password yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-3.

7. PreConditions

1. Aktor telah terdaftar dalam basis data.

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem sesuai role yang dimilikinya.

Program Studi Teknik Informatika	SKPL – PH	23/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.2 Use case Spesification : Logout

1. Brief Description

Use case ini digunakan untuk keluar dari sistem yang digunakan oleh Administrator yang telah terdaftar di dalam basis data.

2. Primary Actor

1. Admin.

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan logout.
2. Sistem memberikan akses keluar dari sistem.
3. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

None.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor keluar dari sistem.

4.1.3 Use case Spesification : Mengedit Profil

1. Brief Description

Use case ini digunakan untuk mengedit profil meliputi nama, email, password dan avatar dari Administrator yang telah terdaftar di dalam basis data.

Program Studi Teknik Informatika	SKPL – PH	24/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan edit profil.
2. Sistem menampilkan form edit profil dan meminta aktor untuk memasukkan data nama, email, password, dan avatar yang ingin di ubah datanya.
3. Aktor menginputkan data.
E-1 Data yang diinputkan aktor salah.
4. Sistem memperbaharui data nama, email, password, dan avatar baru di basis data.

5. Alternative Flow

None.

6. Error Flow

E-1 Data yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-3.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data nama, email, password, dan avatar sudah diperbaharui di basis data.

Program Studi Teknik Informatika	SKPL – PH	25/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.4 Use case Spesification : Mengelola Data

Kelompok Makanan

1. Brief Description

Use case ini digunakan oleh Administrator untuk mengelola data kelompok makanan. Aktor dapat melakukan tambah data, ubah data, tampil data, cari data dan hapus data kelompok makanan.

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data kelompok makanan.
2. Sistem memberikan pilihan untuk melakukan tambah data, ubah data, tampil data, cari data dan hapus data kelompok makanan.
3. Aktor memilih untuk melakukan tambah data kelompok makanan.
A-1 Aktor memilih untuk melakukan ubah data kelompok makanan.
A-2 Aktor memilih untuk melakukan tampil data kelompok makanan.
A-3 Aktor memilih untuk melakukan hapus data kelompok makanan.
A-4 Aktor memilih untuk melakukan cari data kelompok makanan.
4. Aktor menginputkan data kelompok makanan.
5. Aktor meminta sistem untuk menyimpan data kelompok makanan yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – PH	26/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem melakukan pengecekan data kelompok makanan yang telah diinputkan.

E-1 Data kelompok makanan yang diinputkan aktor salah.

7. Sistem menyimpan data kelompok makanan ke basis data.

8. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data kelompok makanan.

1. Sistem menampilkan data kelompok makanan.
2. Aktor memilih data kelompok makanan yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data kelompok yang telah diubah.
4. Sistem melakukan pengecekan terhadap data kelompok yang telah diubah.

E-2 Data kelompok makanan yang telah diubah aktor

Salah.

5. Sistem menyimpan data kelompok makanan yang telah diubah ke basis data.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data kelompok makanan.

1. Sistem menampilkan data kelompok makanan.
2. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan hapus data kelompok makanan.

Program Studi Teknik Informatika	SKPL – PH	27/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan data kelompok makanan yang ingin dihapus.
2. Aktor memilih untuk menghapus data kelompok makanan.
3. Sistem menghapus data kelompok makanan yang dipilih dari basis data.
4. Berlanjut ke Basic Flow langkah ke-8.

A-4 Aktor memilih untuk melakukan cari data kelompok makanan.

1. Sistem menampilkan data kelompok makanan yang dicari.
2. Berlanjut ke Basic Flow langkah ke-8.

6. Error Flow

E-1 Data kelompok yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data kelompok yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data kelompok makanan telah diperbaharui di basis data.

Program Studi Teknik Informatika	SKPL – PH	28/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

**4.1.5 Use case Spesification : Mengelola Data
Daftar Makanan**

1. Brief Description

Use case ini digunakan oleh Administrator untuk mengelola data daftar makanan. Aktor dapat melakukan tambah data, ubah data, tampil data, cari data dan hapus data daftar makanan.

2. Primary Actor

2. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data daftar makanan.
2. Sistem memberikan pilihan untuk melakukan tambah data, ubah data, tampil data, cari data dan hapus data daftar makanan.
3. Aktor memilih untuk melakukan tambah data daftar makanan.
A-1 Aktor memilih untuk melakukan ubah data daftar makanan.
A-2 Aktor memilih untuk melakukan tampil data daftar makanan.
A-3 Aktor memilih untuk melakukan hapus data daftar makanan.
A-4 Aktor memilih untuk melakukan cari data daftar makanan.
4. Aktor menginputkan data daftar makanan.
5. Aktor meminta sistem untuk menyimpan data daftar makanan yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – PH	29/ 52
--	-----------	--------

6. Sistem melakukan pengecekan data daftar makanan yang telah diinputkan.

E-1 Data daftar makanan yang diinputkan aktor salah

7. Sistem menyimpan data daftar makanan ke basis data.

8. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data daftar makanan.

1. Sistem menampilkan data daftar makanan.
2. Aktor memilih data daftar makanan yang akan diubah.
3. Aktor mengubah data dan meminta sistem untuk menyimpan data daftar yang telah diubah.
4. Sistem melakukan pengecekan terhadap data daftar yang telah diubah.

E-2 Data daftar makanan yang telah diubah aktor salah

5. Sistem menyimpan data daftar makanan yang telah diubah ke basis data.
6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data daftar makanan.

1. Sistem menampilkan data daftar makanan.
2. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan hapus data daftar makanan.

Program Studi Teknik Informatika	SKPL – PH	30/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan data daftar makanan yang ingin dihapus.
 2. Aktor memilih untuk menghapus data daftar makanan.
 3. Sistem menghapus data daftar makanan yang dipilih dari basis data.
 4. Berlanjut ke Basic Flow langkah ke-8.
- A-4 Aktor memilih untuk melakukan cari data daftar makanan.
1. Sistem menampilkan data daftar makanan yang dicari.
 2. Berlanjut ke Basic Flow langkah ke-8.

6. Error Flow

E-1 Data daftar makanan yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data daftar makanan yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data daftar makanan telah diperbaharui di basis data.

Program Studi Teknik Informatika	SKPL – PH	31/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.6 Use case Spesification : Mengelola Data Jenis Makanan

1. Brief Description

Use case ini digunakan oleh Administrator untuk mengelola data jenis makanan. Aktor dapat melakukan tambah data, ubah data, tampil data, cari data dan hapus data jenis makanan.

2. Primary Actor

1. Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data jenis makanan.
2. Sistem memberikan pilihan untuk melakukan tambah data, ubah data, tampil data, cari data dan hapus data jenis makanan.
3. Aktor memilih untuk melakukan tambah data jenis makanan.
A-1 Aktor memilih untuk melakukan ubah data jenis makanan.
A-2 Aktor memilih untuk melakukan tampil data jenis makanan.
A-3 Aktor memilih untuk melakukan hapus data jenis makanan.
A-4 Aktor memilih untuk melakukan cari data jenis makanan.
4. Aktor menginputkan data jenis makanan.
5. Aktor meminta sistem untuk menyimpan data jenis makanan yang telah diinputkan.

Program Studi Teknik Informatika	SKPL – PH	32/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Sistem melakukan pengecekan data jenis makanan yang telah diinputkan.

E-1 Data jenis makanan yang diinputkan aktor salah

7. Sistem menyimpan data jenis ke basis data.

8. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data jenis makanan.

1. Sistem menampilkan data jenis makanan.

2. Aktor memilih data jenis makanan yang akan diubah.

3. Aktor mengubah data dan meminta sistem untuk menyimpan data jenis yang telah diubah.

4. Sistem melakukan pengecekan terhadap data jenis yang telah diubah.

E-2 Data jenis makanan yang telah diubah aktor salah

5. Sistem menyimpan data jenis makanan yang telah diubah ke basis data.

6. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan tampil data jenis makanan.

1. Sistem menampilkan data jenis makanan.

2. Berlanjut ke Basic Flow langkah ke-8.

A-3 Aktor memilih untuk melakukan hapus data jenis makanan.

1. Sistem menampilkan data jenis makanan yang ingin dihapus.

Program Studi Teknik Informatika	SKPL – PH	33/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Aktor memilih untuk menghapus data jenis makanan.

3. Sistem menghapus data jenis makanan yang dipilih dari basis data.

4. Berlanjut ke Basic Flow langkah ke-8.

A-4 Aktor memilih untuk melakukan cari data jenis makanan.

1. Sistem menampilkan data daftar makanan yang dicari.

2. Berlanjut ke Basic Flow langkah ke-8.

6. Error Flow

E-1 Data daftar makanan yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.

2. Kembali ke Basic Flow langkah ke-4.

E-2 Data daftar makanan yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.

2. Kembali ke Alternative Flow A-1 langkah ke-3.

7. PreConditions

1. Use case Login telah dilakukan.

2. Aktor telah memasuki sistem.

8. PostConditions

1. Data jenis telah diperbaharui di basis data.

Program Studi Teknik Informatika	SKPL – PH	34/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.7 Use case Spesification : Mengelola Data Pengguna

1. Brief Description

Use case ini digunakan oleh Administrator untuk mengelola data pengguna. Aktor dapat melakukan cari data dan hapus data pengguna.

2. Primary Actor

1. Admin

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola data pengguna.
2. Sistem memberikan pilihan untuk melakukan cari data, tampil data, dan hapus data pengguna.
3. Aktor memilih untuk melakukan cari data pengguna.
A-1 Aktor memilih untuk melakukan tampil data pengguna.
A-1 Aktor memilih untuk melakukan hapus data pengguna.
4. Aktor menginputkan data pengguna yang ingin dicari.
5. Sistem melakukan pengecekan data pengguna yang telah diinputkan.
E-1 Data pengguna yang diinputkan aktor salah
6. Sistem menampilkan data pengguna.
7. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan tampil data daftar makanan

Program Studi Teknik Informatika	SKPL – PH	35/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem menampilkan data daftar makanan.
2. Berlanjut ke Basic Flow langkah ke-8.

A-2 Aktor memilih untuk melakukan hapus data daftar makanan

1. Sistem menampilkan data pengguna yang ingin dihapus.
2. Aktor memilih untuk menghapus data pengguna.
3. Sistem menghapus data pengguna yang dipilih dari basis data.
4. Berlanjut ke Basic Flow langkah ke-8.

A-4 Aktor memilih untuk melakukan cari data pengguna

1. Sistem menampilkan data pengguna yang dicari.
2. Berlanjut ke Basic Flow langkah ke-8.

6. Error Flow

E-1 Data pengguna yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-4.

E-2 Data pengguna yang telah diubah aktor salah

1. Sistem memberi peringatan bahwa data yang diubah salah.
2. Kembali ke Alternative Flow A-1 langkah ke-3.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data pengguna telah diperbaharui di basis data.

Program Studi Teknik Informatika	SKPL – PH	36/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

SISTEM MOBILE

4.1.8 Use case Spesification : Mendaftar

1. Brief Description

Use case ini digunakan oleh aktor untuk melakukan pendaftaran mendaftar (*register*).

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mendaftar.
2. Aktor menginputkan data pengguna.
3. Aktor meminta sistem untuk menyimpan data pengguna yang telah diinputkan.
4. Sistem melakukan pengecekan data pengguna yang telah diinputkan.
E-1 Data pengguna yang diinputkan aktor salah
5. Sistem menyimpan data pengguna ke basis data.
6. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 Data pengguna yang diinputkan aktor salah

1. Sistem memberi peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke-2.

7. PreConditions

1. Aktor belum terdaftar di dalam basis data.

Program Studi Teknik Informatika	SKPL – PH	37/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. Data pengguna telah terdaftar di basis data.

4.1.9 Use case Spesification : Login

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu username dan password yang berupa rangkaian karakter.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login.
2. Sistem menampilkan antarmuka untuk login.
3. Aktor memasukan username dan password.
4. Sistem memeriksa username dan password yang diinputkan actor.

E-1 Password atau username pengguna tidak sesuai.

5. Sistem memberikan akses ke aktor.
6. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 Password atau nama pengguna tidak sesuai

1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai.

Program Studi Teknik Informatika	SKPL – PH	38/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

Aktor masih berada diluar sistem.

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

4.1.10 Use case Spesification : Logout

1. Brief Description

Use Case ini digunakan oleh aktor untuk keluar dari sistem.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan logout.
2. Sistem memberikan akses keluar dari sistem.
3. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

None.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

Program Studi Teknik Informatika	SKPL – PH	39/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.11 Use case Spesification : Menampilkan Informasi Kebutuhan Kalori harian dan Indeks Massa Tubuh

1. Brief Description

Use case ini digunakan oleh Aktor untuk melihat data Informasi Kebutuhan Kalori harian dan Indeks Massa Tubuh dari pengguna.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memasuki sistem.
2. Sistem menampilkan data informasi kebutuhan kalori harian dan indeks massa tubuh dari pengguna.
3. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

None.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

Program Studi Teknik Informatika	SKPL – PH	40/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Data kebutuhan kalori harian dan indeks massa tubuh telah ditampilkan.

4.1.12 Use case Spesification : Mengupdate Informasi Pengguna

1. Brief Description

Use case ini digunakan oleh Pengguna untuk mengupdate informasi pengguna.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih menghitung ulang pada halaman beranda.
2. Sistem menampilkan form untuk mengupdate informasi pengguna.
3. Sistem meminta pengguna untuk menginputkan berat, tinggi, umur, golongan darah dan jenis aktivitas apabila ingin di update.
4. Aktor menginputkan berat, tinggi, umur, golongan darah dan jenis aktivitas apabila ingin di update.
5. Sistem melakukan pengecekan inputan dari pengguna.
E-1 Data yang diinputkan aktor salah
6. Aktor meminta sistem untuk menghitung berdasarkan data yang telah diinputkan.
7. Sistem melakukan kalkulasi berdasarkan inputan pengguna pengguna.

Program Studi Teknik Informatika	SKPL – PH	41/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. Sistem menampilkan indeks massa tubuh berdasarkan hasil kalkulasi.
9. Sistem menyimpan data indeks massa tubuh dari pengguna ke basis data.
10. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 data yang diinputkan salah

1. Sistem menampilkan peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data pengguna telah diperbaharui di basis data.

4.1.13 Use case Spesification : Mengatur Jadwal Makan

1. Brief Description

Use case ini digunakan oleh pengguna untuk mengatur jadwal makan harian pengguna. Sistem akan menampilkan daftar makan pagi, makan siang, makan malam dan snack sesuai dengan tanggal yang diatur. Sistem juga akan menampilkan total dari kalori makanan yang ditambahkan pada daftar. Selain itu sistem akan menampilkan notifikasi untuk makan sesuai jadwal yang ada di database.

2. Primary Actor

Program Studi Teknik Informatika	SKPL – PH	42/ 52
--	-----------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengatur jadwal makan.
2. Sistem menampilkan form jadwal makan.
3. Sistem memberikan pilihan tanggal dan jenis jadwal makan (makan pagi, makan siang, makan malam, dan snack).
4. Aktor memilih tanggal dan jenis jadwal makan.
5. Aktor menambahkan jadwal makan pagi.
 - A-1 Aktor memilih tambah jadwal makan siang.
 - A-2 Aktor memilih tambah jadwal makan malam.
 - A-3 Aktor memilih tambah jadwal Snack.
6. Sistem menampilkan kelompok, daftar dan jenis makanan berdasarkan golongan darah aktor.
7. Aktor memilih tambah jenis makanan tertentu.
8. Sistem menampilkan form detail makanan yang dipilih aktor dan meminta aktor menginputkan porsi.
9. Aktor menginputkan porsi untuk jenis makanan yang dipilih.
10. Sistem melakukan kalkulasi kalori, protein, lemak, karbohidrat berdasarkan porsi yang diinputkan aktor.
11. Aktor meminta sistem untuk menyimpan jenis makanan yang dipilih ke jadwal makan pagi.
 - E-1 Total kalori sudah melebihi kebutuhan kalori harian.

Program Studi Teknik Informatika	SKPL – PH	43/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

12. Sistem menyimpan data ke basis data.

13. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih tambah jadwal makan siang

1. Berlanjut ke Basic Flow langkah ke-6 sampai 10.

2. Aktor meminta sistem untuk menyimpan jenis makanan yang dipilih ke jadwal makan siang.

3. Sistem menyimpan data ke basis data.

4. Use case selesai.

A-2 Aktor memilih tambah jadwal makan malam

1. Berlanjut ke Basic Flow langkah ke-6 sampai 10.

2. Aktor meminta sistem untuk menyimpan jenis makanan yang dipilih ke jadwal makan malam.

3. Sistem menyimpan data ke basis data.

4. Use case selesai.

A-3 Aktor memilih tambah jadwal snack

1. Berlanjut ke Basic Flow langkah ke-6 sampai 10.

2. Aktor meminta sistem untuk menyimpan jenis makanan yang dipilih ke jadwal snack.

3. Sistem menyimpan data ke basis data.

4. Use case selesai.

6. Error Flow

E-1 Total kalori sudah melebihi kebutuhan kalori harian

1. Sistem menampilkan dialog peringatan bahwa total kalori akan melebihi jika tetap ingin ditambahkan.

Program Studi Teknik Informatika	SKPL – PH	44/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Jika Aktor memilih membatalkan Kembali ke Basic Flow langkah ke 9

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data jadwal makan telah diperbaharui di database.

4.1.14 Use case Spesification : Menghitung Indeks Massa Tubuh

1. Brief Description

Use case ini digunakan oleh Pengguna untuk mengetahui indeks massa tubuh pengguna.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih hitung indeks massa tubuh.
2. Sistem menampilkan form untuk menghitung indeks massa tubuh.
3. Sistem meminta pengguna untuk menginputkan berat, tinggi, jenis kelamin.
4. Aktor menginputkan berat, tinggi, jenis kelamin.
5. Sistem melakukan pengecekan inputan dari pengguna.

E-1 Data yang diinputkan salah

Program Studi Teknik Informatika	SKPL – PH	45/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Aktor meminta sistem untuk menghitung berdasarkan data yang telah diinputkan.
7. Sistem melakukan kalkulasi indeks massa tubuh pengguna.
8. Sistem menampilkan indeks massa tubuh berdasarkan hasil kalkulasi.
9. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 data yang diinputkan salah

1. Sistem menampilkan peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data indeks massa tubuh telah dihitung dan ditampilkan.

4.1.15 Use case Spesification : Menghitung Kebutuhan Kalori Harian

1. Brief Description

Use case ini digunakan oleh Pengguna untuk mengetahui kebutuhan kalori harian pengguna.

2. Primary Actor

2. Pengguna.

3. Supporting Actor

None.

Program Studi Teknik Informatika	SKPL – PH	46/ 52
--	-----------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih hitung kebutuhan kalori harian.
2. Sistem menampilkan form untuk menghitung kebutuhan kalori harian.
3. Sistem meminta pengguna untuk menginputkan berat, tinggi, umur, jenis kelamin dan level aktivitas.
4. Aktor menginputkan berat, tinggi, jenis kelamin, umur dan level aktivitas.
5. Sistem melakukan pengecekan inputan dari pengguna.
E-1 Data yang diinputkan salah
6. Aktor meminta sistem untuk menghitung berdasarkan data yang telah diinputkan.
7. Sistem melakukan kalkulasi kebutuhan kalori pengguna.
8. Sistem menampilkan kebutuhan kalori harian berdasarkan hasil kalkulasi.
9. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 data yang diinputkan salah

1. Sistem menampilkan peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – PH	47/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. Data kebutuhan kalori telah dihitung dan ditampilkan.

4.1.16 Use case Spesification : Mengubah Akun

1. Brief Description

Use case ini digunakan oleh Pengguna untuk mengubah informasi akun pengguna.

2. Primary Actor

1. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih pengaturan akun.
2. Sistem menampilkan form untuk mengupdate akun pengguna.
3. Sistem meminta pengguna untuk menginputkan nama pengguna, username, dan password apabila ingin di update.
4. Aktor menginputkan nama pengguna, username, dan password apabila ingin di update.
5. Sistem melakukan pengecekan inputan dari pengguna.
E-1 Data yang diinputkan aktor salah
6. Aktor meminta sistem untuk menghitung berdasarkan data yang telah diinputkan.
7. Sistem menyimpan data akun dari pengguna ke basis data.
8. Use case selesai.

Program Studi Teknik Informatika	SKPL – PH	48/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Alternative Flow

None.

6. Error Flow

E-1 data yang diinputkan salah

1. Sistem menampilkan peringatan bahwa data yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data akun pengguna telah diperbaharui di basis data.

4.1.17 Menampilkan Makanan Berdasarkan Golongan Darah

1. Brief Description

Use case ini digunakan oleh pengguna untuk menampilkan makanan berdasarkan golongan darah.

2. Primary Actor

2. Pengguna.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk menambahkan jadwal makan berdasarkan golongan darah.
2. Sistem menampilkan makanan sesuai dengan kelompok, daftar, dan jenis makanan berdasarkan golongan darah.

Program Studi Teknik Informatika	SKPL – PH	49/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

none

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data kelompok, daftar dan jenis makanan telah diperbaharui di database.

4.1.18 Memberikan Notifikasi Jadwal Makan

1. Brief Description

Use case ini digunakan oleh pengguna untuk me.

2. Primary Actor

1. Sistem.

3. Supporting Actor

1. Aktor.

4. Basic Flow

1. Use Case ini dimulai ketika aktor sudah mengelola jadwal makan sudah.
2. Sistem menampilkan notifikasi untuk mengingatkan waktu makan.
3. Use case selesai.

5. Alternative Flow

None.

6. Error Flow

None.

7. PreConditions

Program Studi Teknik Informatika	SKPL – PH	50/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use case Mengelola Jadwal Makan telah dilakukan.

8. PostConditions

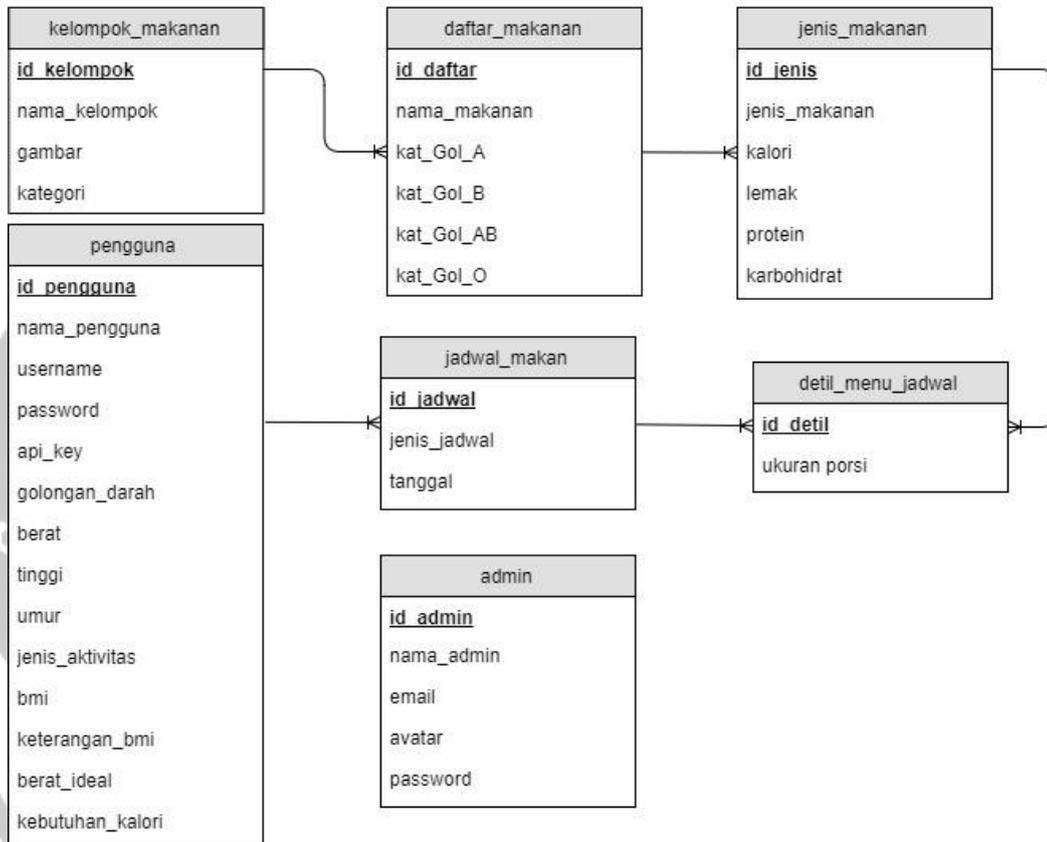
1. Memberikan notifikasi jadwal makan.



Program Studi Teknik Informatika	SKPL – PH	51/ 52
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5 ERD

Berikut adalah ERD dari perangkat lunak PH yang akan dibangun dapat dilihat pada gambar 5.1



Gambar 5. 1 Entity Relationship Diagram Perangkat Lunak PH