

BAB VI

PENUTUP

Pada bab penutup ini akan diberikan kesimpulan dan saran yang didapatkan selama pembuatan tugas akhir ini.

6.1. Kesimpulan

Berdasarkan pembahasan dari bab-bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi sistem pakar mendiagnosa penyakit pada organ pankreas manusia berhasil dibangun menggunakan metode *Certainty Factor*.
2. Aplikasi ini bermanfaat bagi masyarakat umum dalam mendiagnosa penyakit pada organ pankreas manusia dan menambah pengetahuan bagi masyarakat umum mengenai macam-macam penyakit pada organ pankreas.

6.2. Saran

Dari proses analisis, perancangan, implementasi, dan pengujian sistem pakar pada pembuatan tugas akhir ini, didapatkan beberapa saran untuk pengembangan lebih lanjut untuk sistem pakar yang dibangun untuk tugas akhir ini, antara lain:

1. Diharapkan sistem ini dapat dikembangkan untuk penyakit selain organ pankreas manusia.
2. Penambahan aturan-aturan baru mengenai penyakit pada organ pankreas manusia pada basis pengetahuan untuk meningkatkan akurasi dalam mendiagnosa.

DAFTAR PUSTAKA

- Akmal, F. & Winiarti, S., 2014. Sistem Pakar Untuk Mendiagnosa Penyakit Lambung Dengan Implementasi Metode CBR (Case Based Reasoning) Berbasis Web. *Jurnal Sarjana Teknik Informatika*, 2(1), pp. 790-800.
- Alfaris, S., 2014. Sistem Pakar Untuk Mendiagnosa Penyakit Polip Nasi (Polip Hidung) Menggunakan Metode Certainty Factory. *Pelita Informatika Budi Darma*, 7(2), pp. 147-152.
- Amrullah, A. J. A. & Ekojono, 2015. Sistem Pakar Diagnosa Penyakit Paru-Paru Dengan Metode Forward Chaining. *Jurnal Informatika Polinema*, 2(1), pp. 13-17.
- Arif, S. N., 2011. Implementasi Rule-Based Expert Systems Dalam Mendeteksi Kerusakan Sistem Jaringan Komputer Dengan Metode Backward Chaining. *Jurnal SAINTIKOM*, 10(2), pp. 81-93.
- Dahria, M., 2011. Pengembangan Sistem Pakar Dalam Membangun Suatu Aplikasi. *Jurnal SAINTIKOM*, 10(3), pp. 199-205.
- Developer, A., 2013. *Download Android Studio and SDK Tools | Android Studio..* [Online]
Available at:
<https://developer.android.com/studio/index.html>
[Accessed 13 November 2016].
- Fadhilah, A. N., Destiani, D. & Dhamiri, D. J., 2012. Perancangan Aplikasi Sistem Pakar Penyakit Kulit Pada Anak Dengan Metode Expert System Development Life

- Cycle. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*, 9(13), pp. 1-7.
- Fadli, A., Retno, A. & Ririd, T. H., 2016. Sistem Pakar Diagnosa 33 Macam Penyakit Kulit Dan Kelamin Dengan Metode Fuzzy Inference Tsukamoto. *Jurnal Informatika Polinema*, 2(3), pp. 132-137.
- Gaol, S. D. L., 2013. Sistem Pakar Mendeteksi Gizi Buruk Pada Balita Berbasis Web Dengan Menggunakan Metode Certainty Factor. *Pelita Informatika Budi Darma*, 5(1), pp. 126-133.
- Halim, S. & Hansun, S., 2015. Penerapan Metode Certainty Factor Dalam Sistem Pakar Pendeteksi Resiko Osteoporosis dan Osteoarthritis. *ULTIMA Computing*, 7(2), pp. 59-69.
- Halosehat, 2016. *5 Gangguan Kelenjar Pankreas - Penyebab dan Gejalanya*. [Online]
Available at: <http://halosehat.com/tips-kesehatan/kesehatan-pencernaan/gangguan-kelenjar-pankreas>
[Accessed 10 Maret 2017].
- Istiqomah, Y. N. & Fadlil, A., 2013. Sistem Pakar Untuk Mendiagnosa Penyakit Saluran Pencernaan Menggunakan Metode Dempster Shafer. *Jurnal Sarjana Teknik Informatika*, 1(1), pp. 32-41.
- Kaswidjanti, W., 2011. Implementasi Mesin Inferensi Fuzzy (Studi Kasus Sistem Pakar Untuk Mendiagnosa Penyakit Tanaman Cabe). *TELEMATIKA*, 7(2), pp. 129-138.

Mahmud, A., Rokhman, M. M. & Prasetyo, A. E., 2016. Rancang Bangun Sistem Pakar Untuk Mendiagnosis Tanaman Cabai Menggunakan Metode Bayes. *Jurnal Rekayasa Dan Manajemen Sistem Informasi*, 2(2), pp. 85-90.

Natalina, B. D., 2013. Sistem Pakar Untuk Mediagnosa Penyakit Alopesia Pada Manusia Dengan Metode Certainty Factor. *Pelita Informatika Budi Darma*, 4(3), pp. 9-14.

Nuraisyah, S., Wijaya, I. D. & Astiningrum, M., 2015. Sistem Pakar Untuk Mendiagnosa Penyakit Kusta Menggunakan Metode Backward Chaining. *Jurnal Informatika Polinema*, 1(4), pp. 55-61.

Ongko, E., 2013. Perancangan Sistem Pakar Diagnosa Penyakit Pada Mata. *Jurnal TIME*, 2(2), pp. 10-17.

Pendidikanku, 2015. *Fungsi Pankreas*. [Online]

Available at:

<http://www.pendidikanku.org/2015/07/pengertian-pankreas-fungsi-dan.html>

[Accessed 13 November 2016].

Rachmawati, Damiri, D. J. & Susanto, A., 2012. Aplikasi Sistem Pakar Diagnosis Penyakit Asma. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*, 9(8), pp. 1-7.

Raharjo, J. S. D., Damiyana, D. & Hidayatullah, M., 2016. Sistem Pakar Diagnosa Penyakit Lambung Dengan Metode Forward Chaining Berbasis Android. *Jurnal SISFOTEK GLOBAL*, 6(2), pp. 1-8.

- Ritonga, N. A., 2013. Sistem Pakar Mendiagnosa Penyakit Lambung Menggunakan Metode Certainty Factor. *Pelita Informatika Budi Darma*, 5(1), pp. 157-161.
- Sari, N. A., 2013. Sistem Pakar Mendiagnosa Penyakit Demam Berdarah Menggunakan Metode Certainty Factor. *Pelita Informatika Budi Darma*, 4(3), pp. 100-103.
- Septiana, L., 2016. Perancangan Sistem Pakar Diagnosa Penyakit Ispa Dengan Metode Certainty Factor Berbasis Android. *Jurnal Techno Nusa Mandiri*, 13(2), pp. 1-8.
- Tempo, 2011. *Mengapa Kanker Pankreas Ganas Pada Jobs..* [Online]
Available at:
<https://m.tempo.co/read/news/2011/10/08/060360462/mengapa-kanker-pankreas-ganas-pada-jobs>
[Accessed 13 November 2016].
- Trisulistyo, F. & Noviyanto, F., 2014. Sistem Pakar Untuk Mendiagnosa Penyakit Hepatitis Dengan Metode Fuzzy Tsukamoto. *Jurnal Sarjana Teknik Informatika*, 2(2), pp. 1160-1166.
- Wardani, A. K., Santoso, N. & Asmara, R. A., 2014. Sistem Pakar Diagnosa Penyakit Diabetes Melitus. *Jurnal Informatika Polinema*, 1(1), pp. 65-71.
- Widiastuti, W., Destiani, D. & Damiri, D. J., 2012. Aplikasi Sistem Pakar Deteksi Dini Pada Penyakit Tuberkulosis. *Jurnal Algoritma Sekolah Tinggi Teknologi Garut*, 9(6), pp. 1-10.
- Winiarti, S., 2008. Pemanfaatan Teorema Bayes Dalam Penentuan Penyakit THT. *Jurnal Informatika*, 2(2), pp. 189-199.

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SiPaPa

(Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Lukas Kurniawan/130707190

Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta
2017

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		DPPL-SiPaPa		1/83
	Fakultas Teknologi Industri	Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh	LK							
Diperiksa oleh	MRT							
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi



Daftar Isi

1.	Pendahuluan	6
1.1.	Tujuan	6
1.2.	Lingkup Masalah	6
1.3.	Definisi dan Akronim	7
1.4.	Referensi	8
2.	Perancangan Sistem	9
2.1.	Perancangan Arsitektur	9
2.2.	Perancangan Rinci	10
2.2.1.	Sequence Diagram	10
2.2.1.1	Fungsi Login	10
2.2.1.2	Fungsi Tambah Data Penyakit.....	11
2.2.1.3	Fungsi Ubah Data Penyakit.....	12
2.2.1.4	Fungsi Hapus Data Penyakit.....	13
2.2.1.5	Fungsi Tampil Data Penyakit.....	14
2.2.1.6	Fungsi Tambah Data User.....	15
2.2.1.7	Fungsi Ubah Data User.....	15
2.2.1.8	Fungsi Hapus Data User.....	16
2.2.1.9	Fungsi Tampil Data User.....	16
2.2.1.10	Fungsi Tambah Data Gejala.....	17
2.2.1.11	Fungsi Ubah Data Gejala.....	18
2.2.1.12	Fungsi Hapus Data Gejala.....	19
2.2.1.13	Fungsi Tampil Data Gejala.....	20
2.2.1.14	Fungsi Tambah Data Solusi.....	21
2.2.1.15	Fungsi Ubah Data Solusi.....	22
2.2.1.16	Fungsi Hapus Data Solusi.....	23
2.2.1.17	Fungsi Tampil Data Solusi.....	24
2.2.1.18	Fungsi Informasi.....	25
2.2.1.19	Fungsi Diagnosa.....	25
2.2.1.20	Fungsi Cari Data Penyakit.....	26
2.2.1.21	Fungsi Cari Data Gejala.....	26
2.2.1.22	Fungsi Cari Data Solusi.....	27
2.2.1.23	Fungsi Cari Data User.....	28
2.2.2.	Class Diagram	29
2.2.3.	Spesifikasi Deskripsi Kelas Diagram	30
3.	Perancangan Data	50
3.1	Dekomposisi Data	50
4.	Deskripsi Perancangan Antar Muka	59

Daftar Gambar

Gambar Physical Data Model.....	58
Gambar Antarmuka Halaman Login.....	59
Gambar Antarmuka Halaman Menu Admin.....	60
Gambar Antarmuka Halaman Menu Pakar.....	61
Gambar Antarmuka Halaman Menu User.....	62
Gambar Antarmuka Halaman Kelola Data Penyakit.....	63
Gambar Antarmuka Halaman Tambah Ubah Data Penyakit.....	64
Gambar Antarmuka Halaman Kelola Data Gejala.....	65
Gambar Antarmuka Halaman Tambah Data Gejala.....	66
Gambar Antarmuka Halaman Ubah Data Gejala.....	67
Gambar Antarmuka Halaman Kelola Data Solusi.....	68
Gambar Antarmuka Halaman Tambah Ubah Data Solusi...	69
Gambar Antarmuka Halaman Kelola Data User.....	70
Gambar Antarmuka Halaman Tambah Ubah Data User.....	71
Gambar Antarmuka Halaman Kelola Riwayat Untuk Admin dan Pakar.....	72
Gambar Antarmuka Halaman Kelola Riwayat Untuk User.	73
Gambar Antarmuka Halaman Informasi.....	73
Gambar Antarmuka Halaman Diagnosa Pertama.....	74
Gambar Antarmuka Halaman Diagnosa Kedua.....	75
Gambar Antarmuka Halaman Hasil Diagnosa.....	76
Gambar Antarmuka Halaman Kategori.....	76
Gambar Antarmuka Halaman Cari Data Penyakit.....	77
Gambar Antarmuka Halaman Cari Data Gejala.....	78
Gambar Antarmuka Halaman Cari Data Solusi.....	79
Gambar Antarmuka Halaman Cari Data Riwayat.....	80
Gambar Antarmuka Halaman Cari Data User.....	81
Gambar Antarmuka Halaman Daftar.....	82
Gambar Antarmuka Halaman Ubah Password.....	82

1. Pendahuluan

1.1. Tujuan

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

1.2. Lingkup Masalah

Perangkat Lunak SiPaPa dikembangkan dengan tujuan untuk:

1. Memudahkan pengguna dalam mendapatkan informasi mengenai penyakit pada organ pankreas manusia.
2. Melakukan pengelolaan seperti penambahan, pengubahan, penghapusan terhadap data penyakit, gejala, solusi, pengguna yang dikelola oleh admin dan pakar.
3. Memudahkan pengguna mengetahui solusi mengenai penyakit pada organ pankreas manusia.

Dan berjalan pada lingkungan dengan platform *Mobile*.

1.3. Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Merupakan deskripsi perancangan perangkat lunak yang akan dikembangkan.
Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor (SiPaPa)	Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia adalah sebuah sistem pakar yang digunakan untuk mendiagnosa, menampilkan informasi, dan solusi mengenai penyakit yang terdapat pada organ pankreas manusia
Aplikasi Mobile	Aplikasi yang berjalan pada perangkat mobile.
Hak Akses	Hak untuk masuk dalam mengelola data pada SiPaPa
Login	Proses autentifikasi untuk memastikan bahwa yang menggunakan aplikasi adalah pengguna yang memiliki hak akses.
Data	Kumpulan fakta yang belum memiliki arti. Dalam kasus ini yang berhubungan dengan

	mendiagnosa penyakit pada SiPaPa.
Basis Data	Tempat penyimpanan data.
Sistem Operasi / OS	Perangkat lunak <i>mobile</i> yang bertugas melakukan kontrol dan manajemen perangkat keras dan juga operasi-operasi dasar sistem.

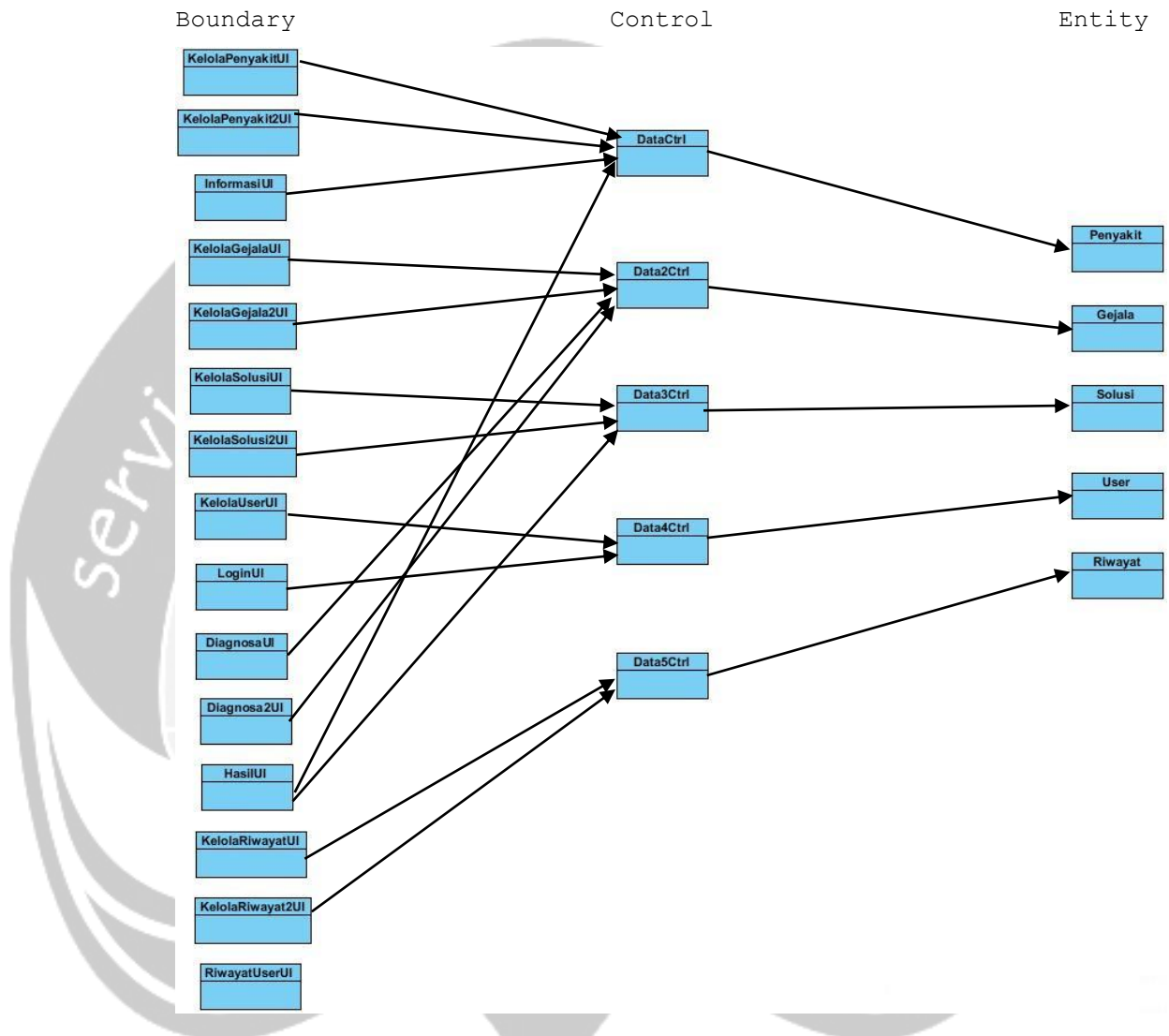
1.4. Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Lukas Kurniawan, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ILUSI*, Universitas Atma Jaya Yogyakarta.
2. Lukas Kurniawan, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) SIPA*, Universitas Atma Jaya Yogyakarta.
3. Aditya Budiman, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) BRsErs*, Universitas Atma Jaya Yogyakarta.

2. Perancangan Sistem

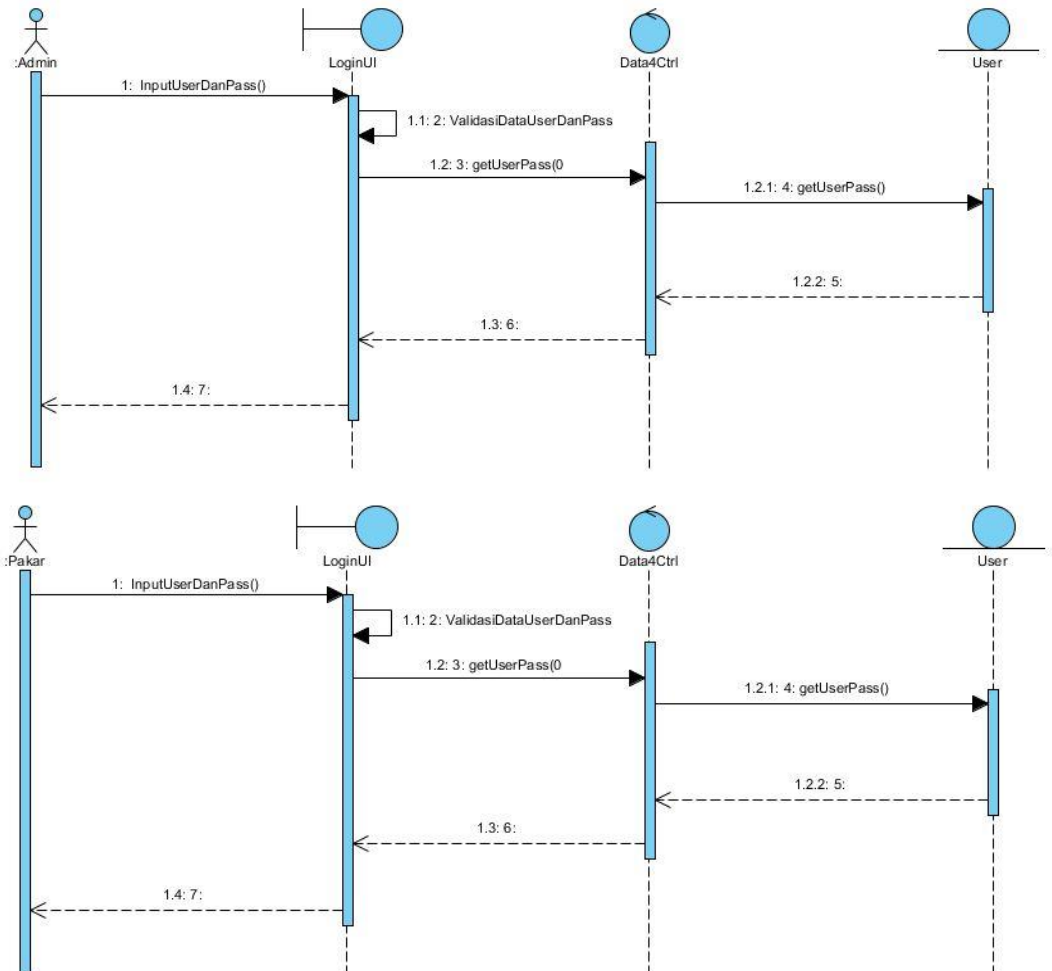
2.1. Perancangan Arsitektur

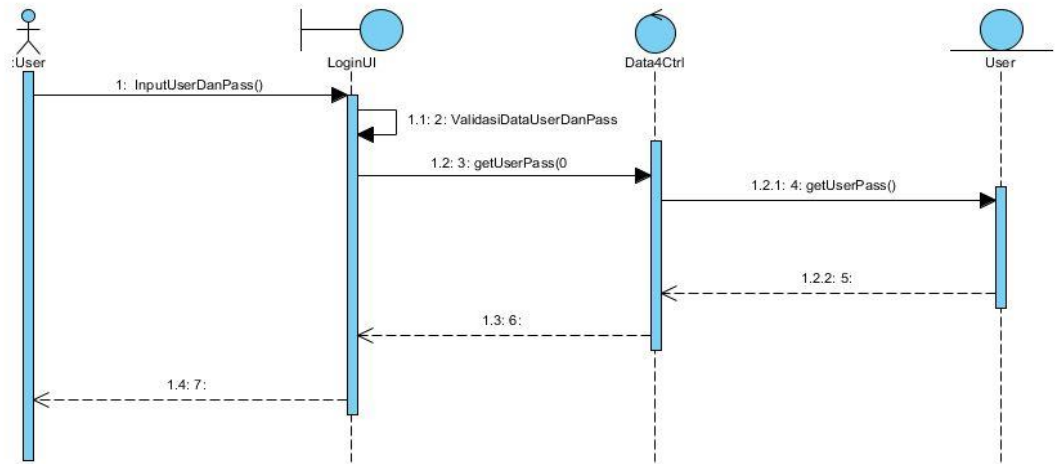


2.2. Perancangan Rinci

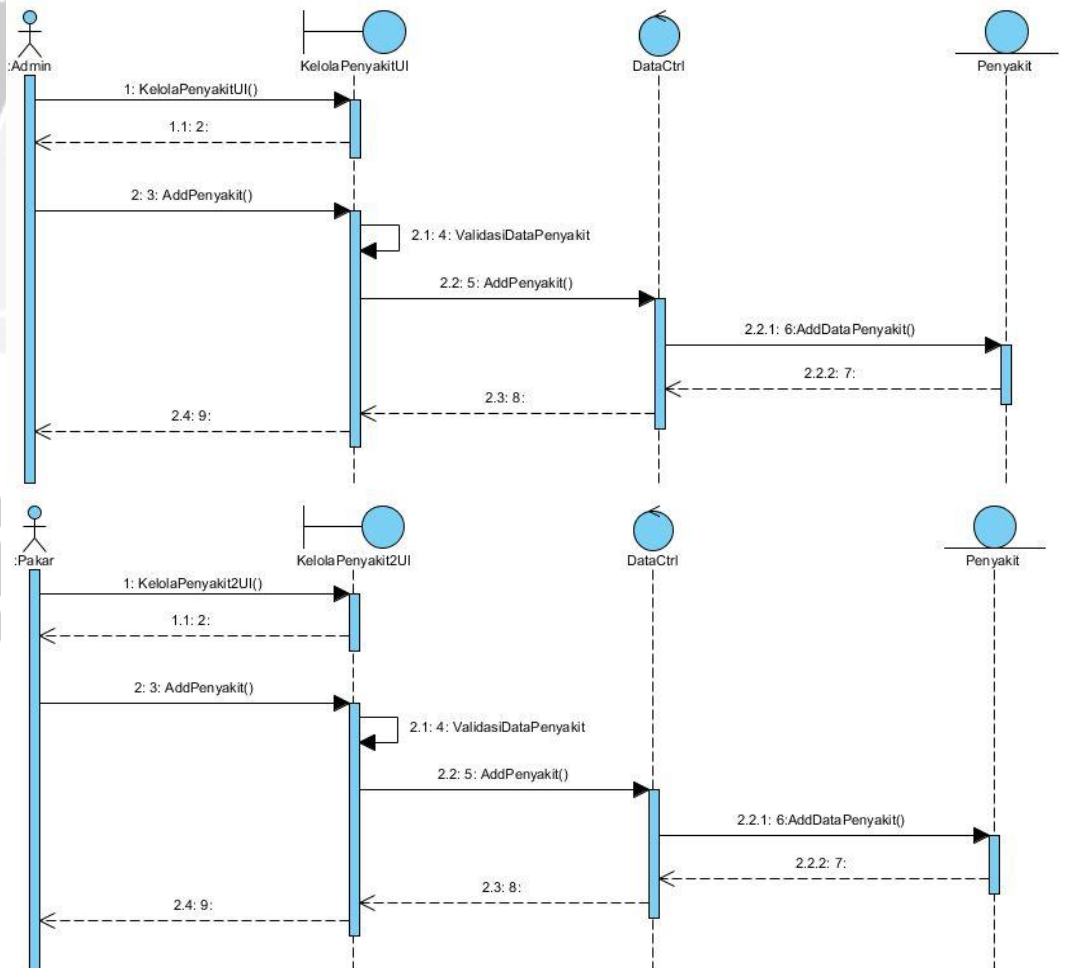
2.2.1. Sequence Diagram

2.2.1.1. Fungsi Login

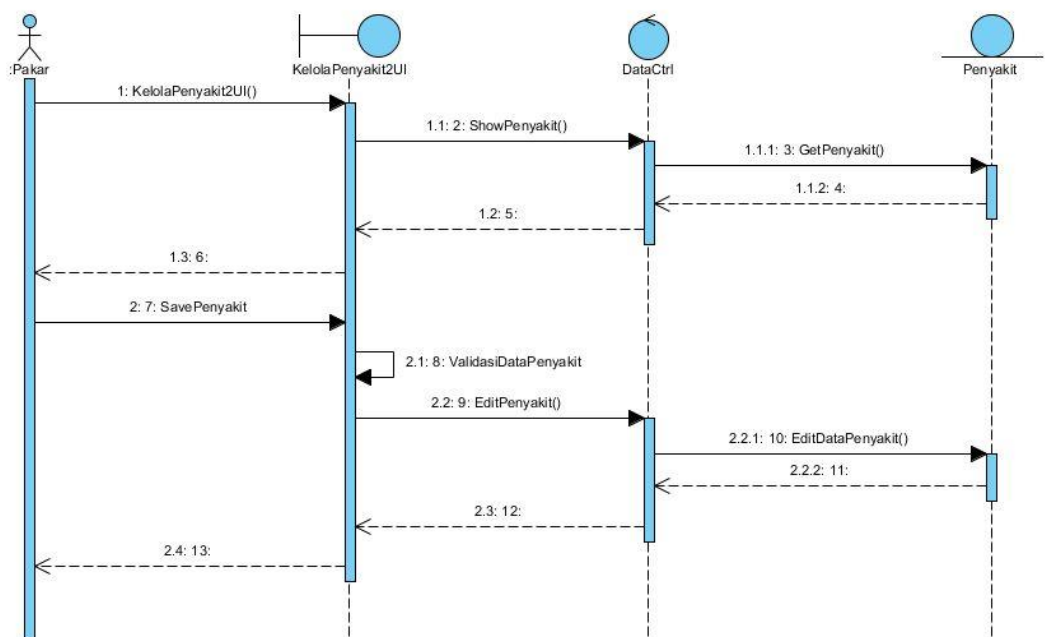
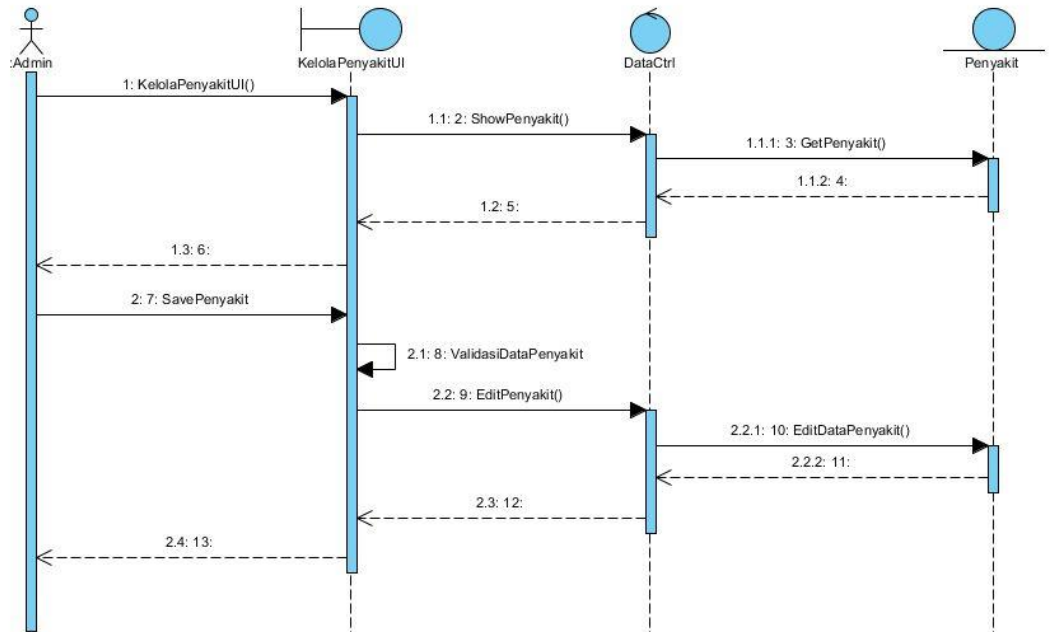




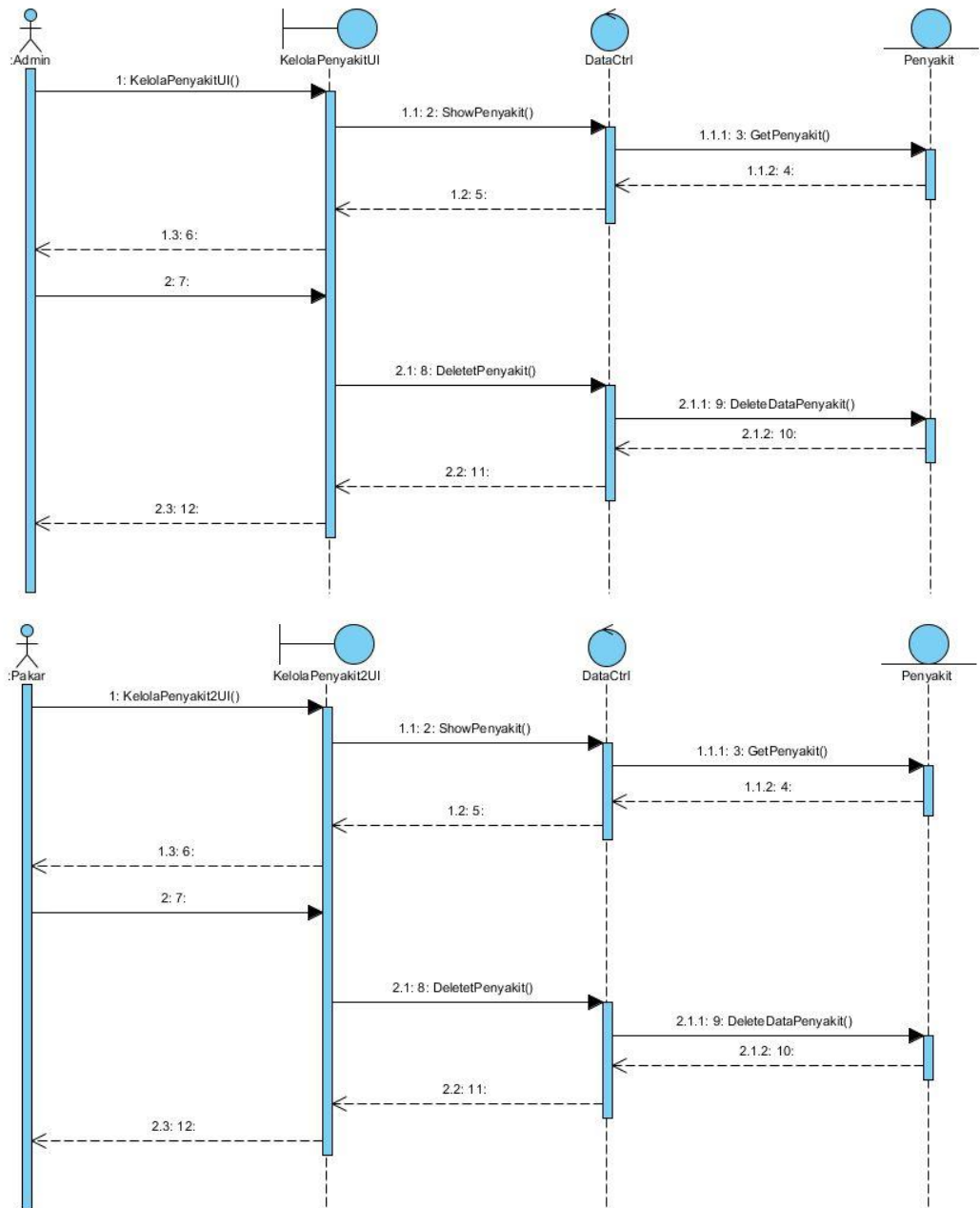
2.2.1.2. Fungsi Tambah Data Penyakit



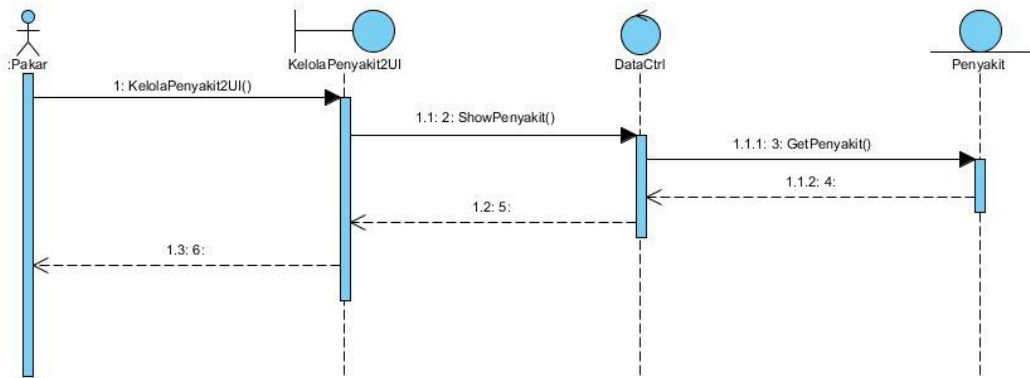
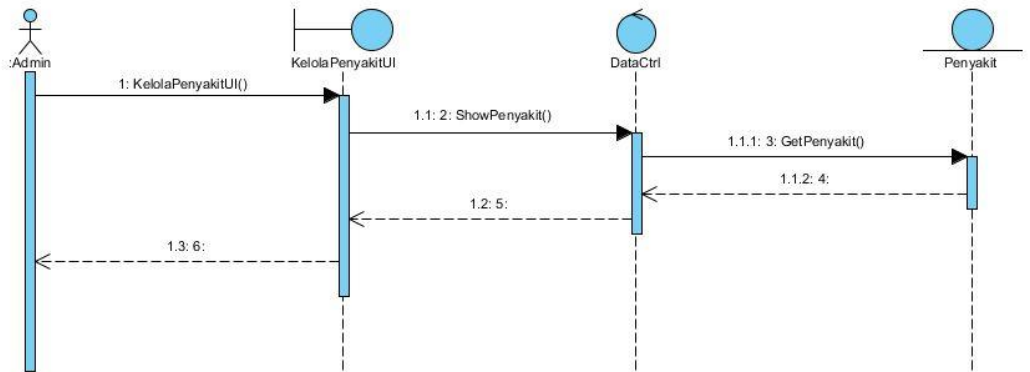
2.2.1.3. Fungsi Ubah Data Penyakit



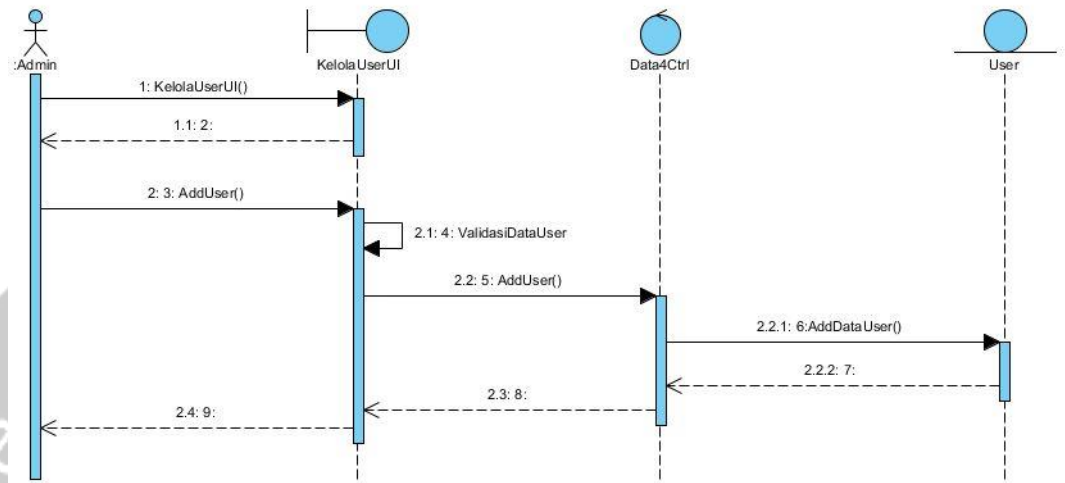
2.2.1.4. Fungsi Hapus Data Penyakit



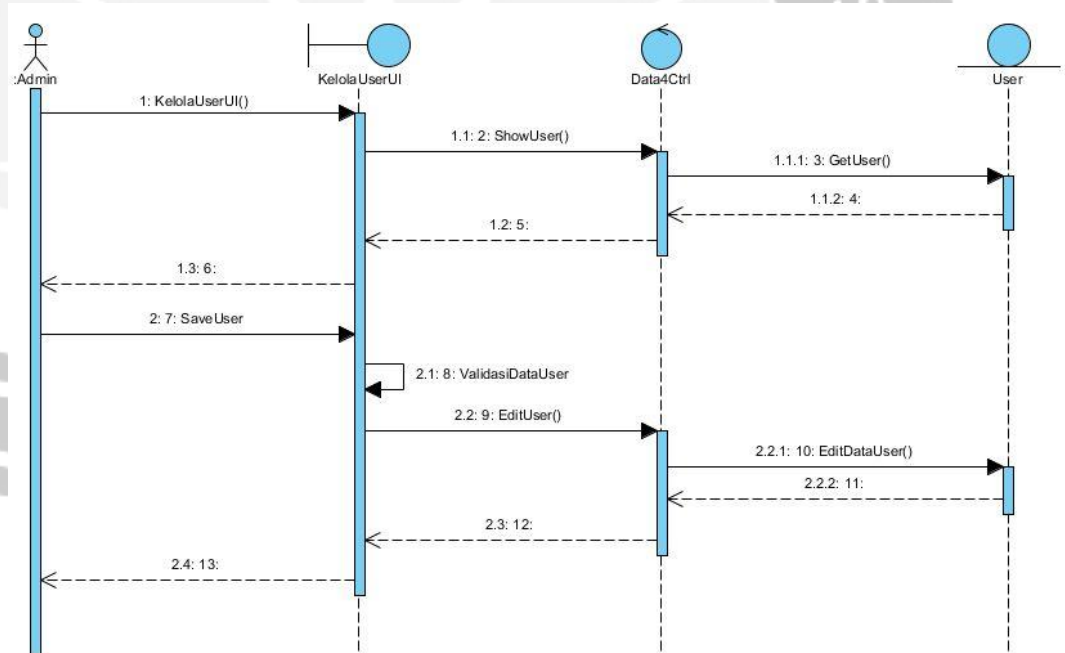
2.2.1.5. Fungsi Tampil Data Penyakit



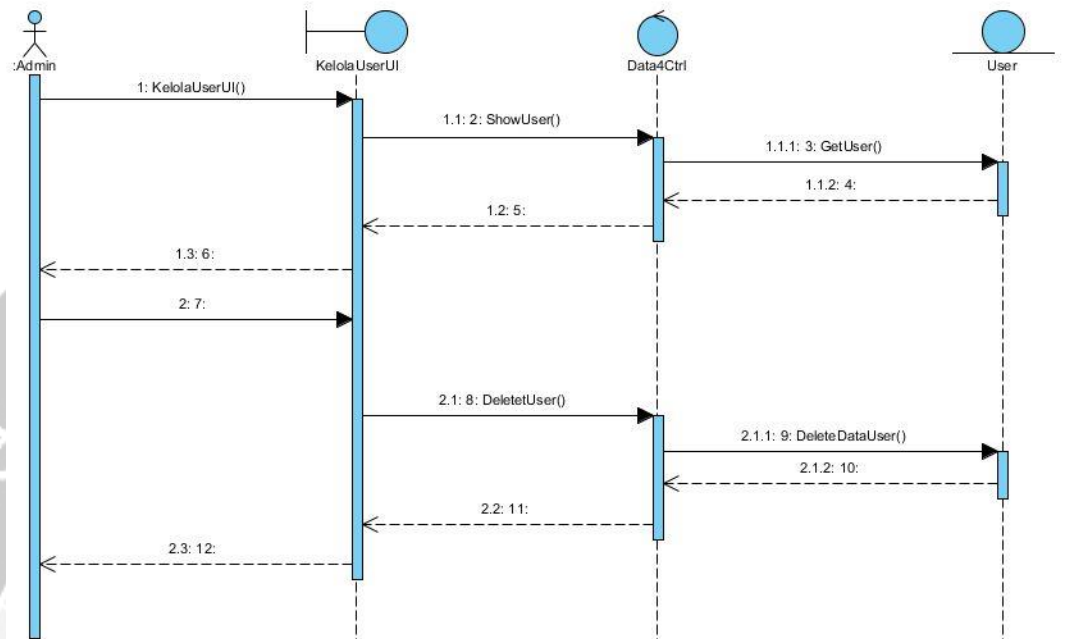
2.2.1.6. Fungsi Tambah Data User



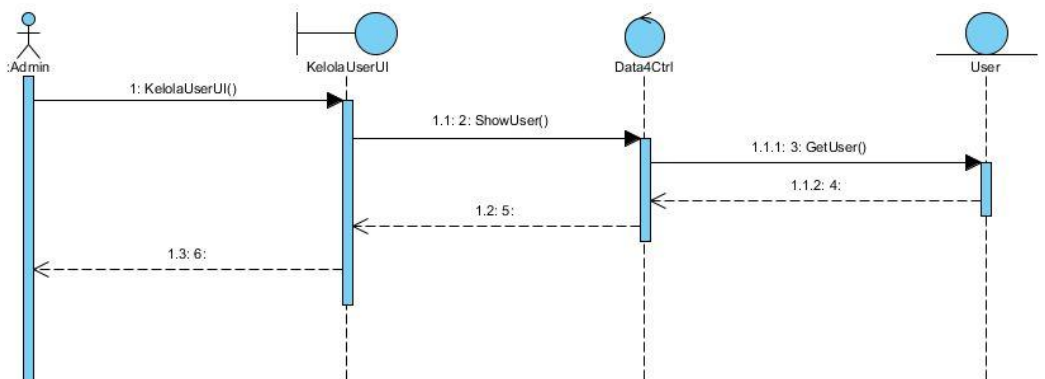
2.2.1.7. Fungsi Ubah Data User



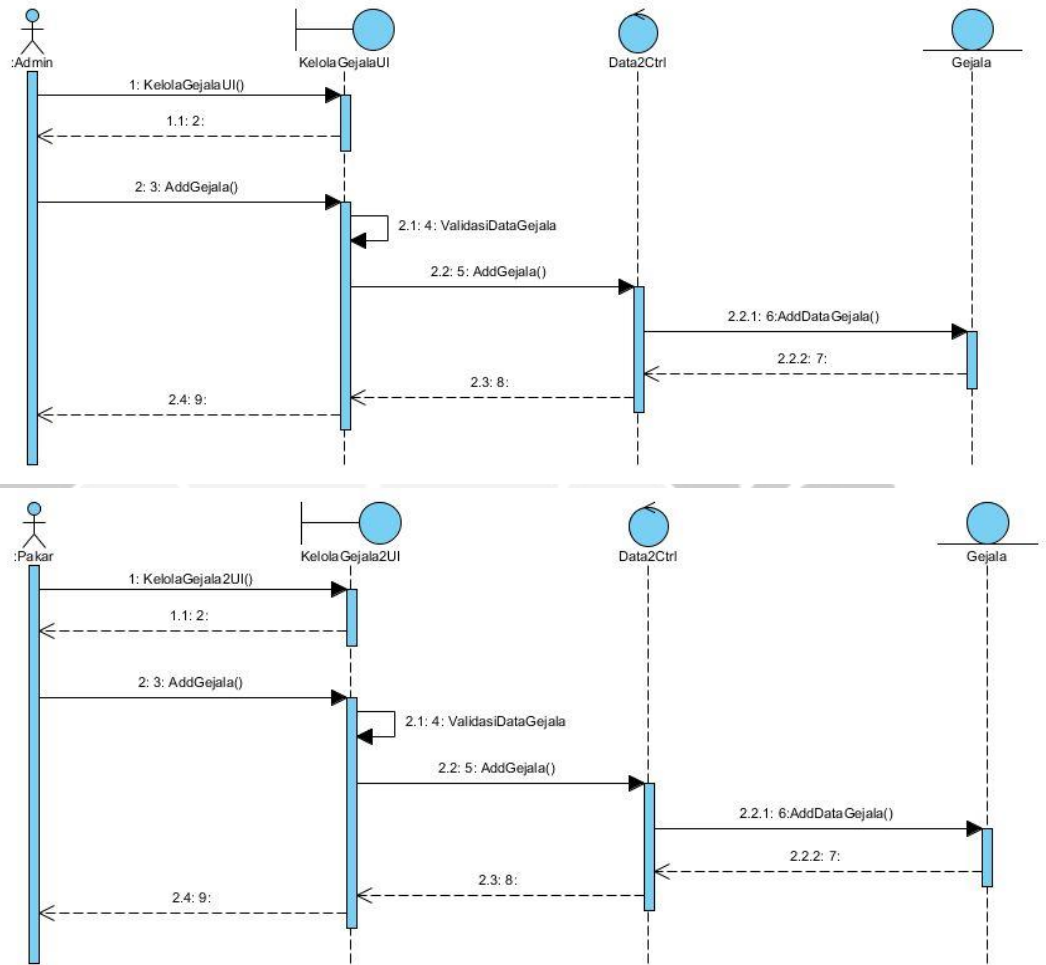
2.2.1.8. Fungsi Hapus Data User



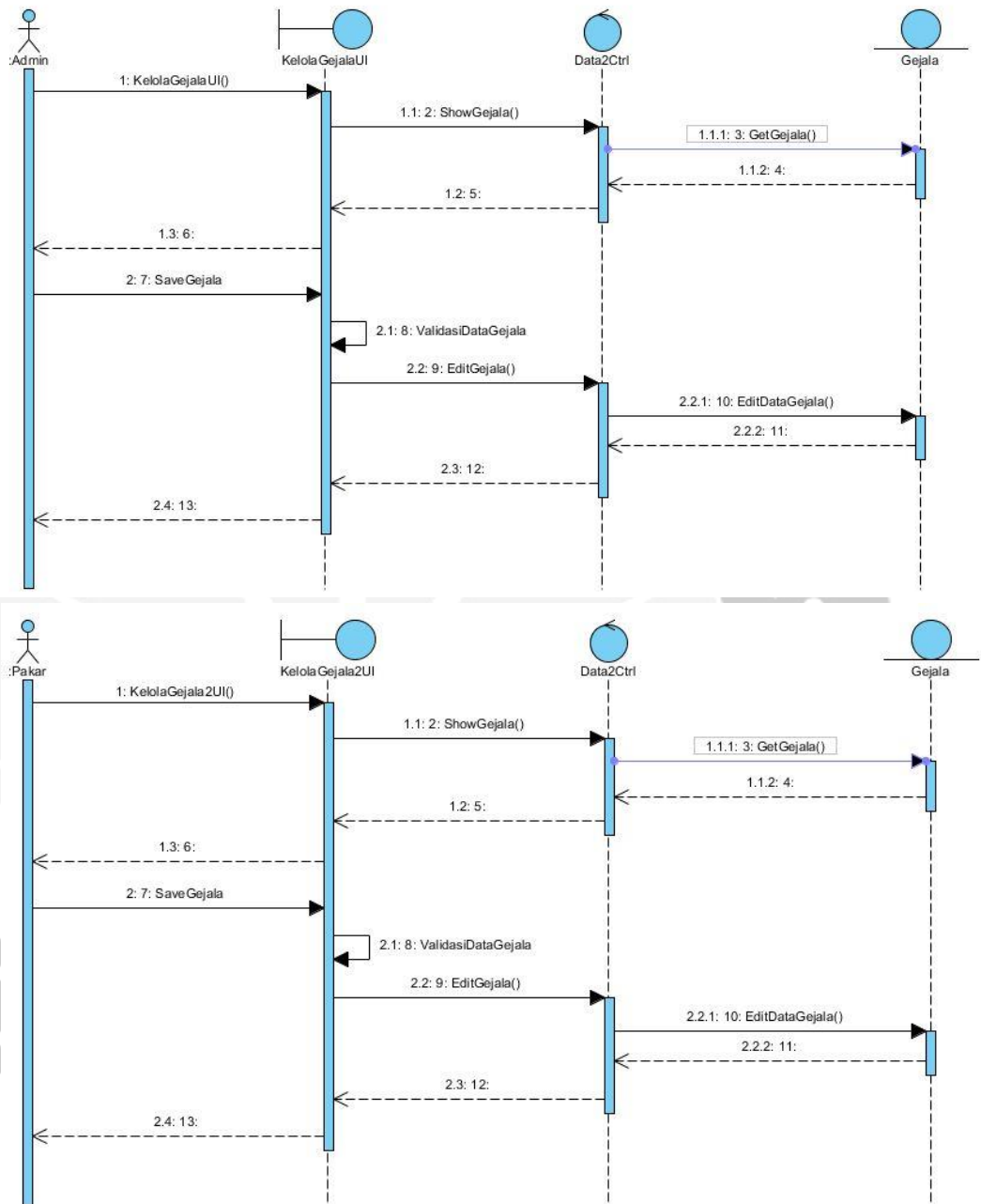
2.2.1.9. Fungsi Tampil Data User



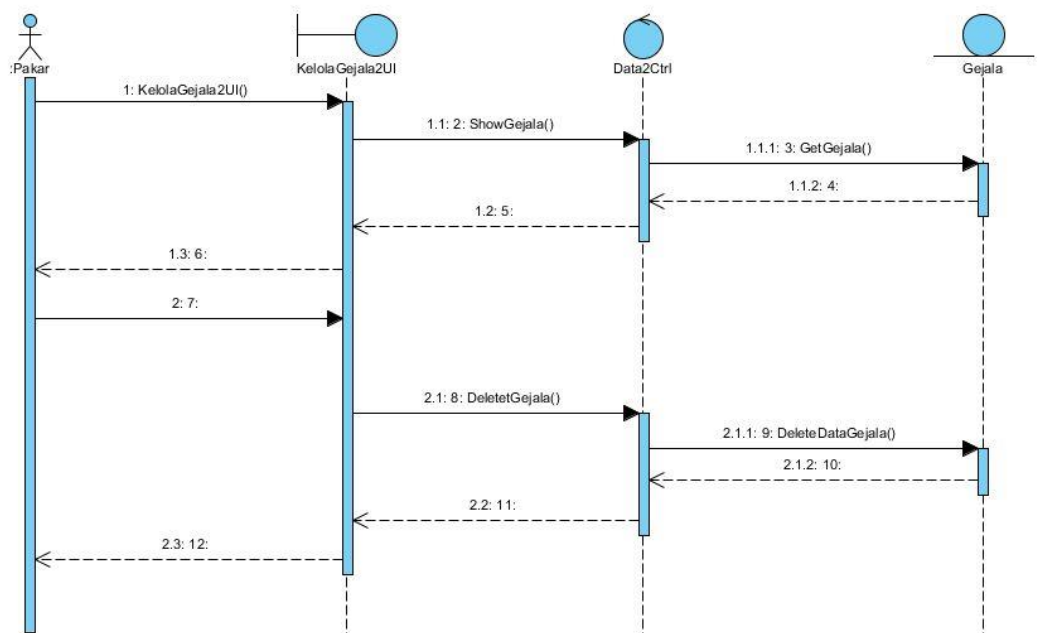
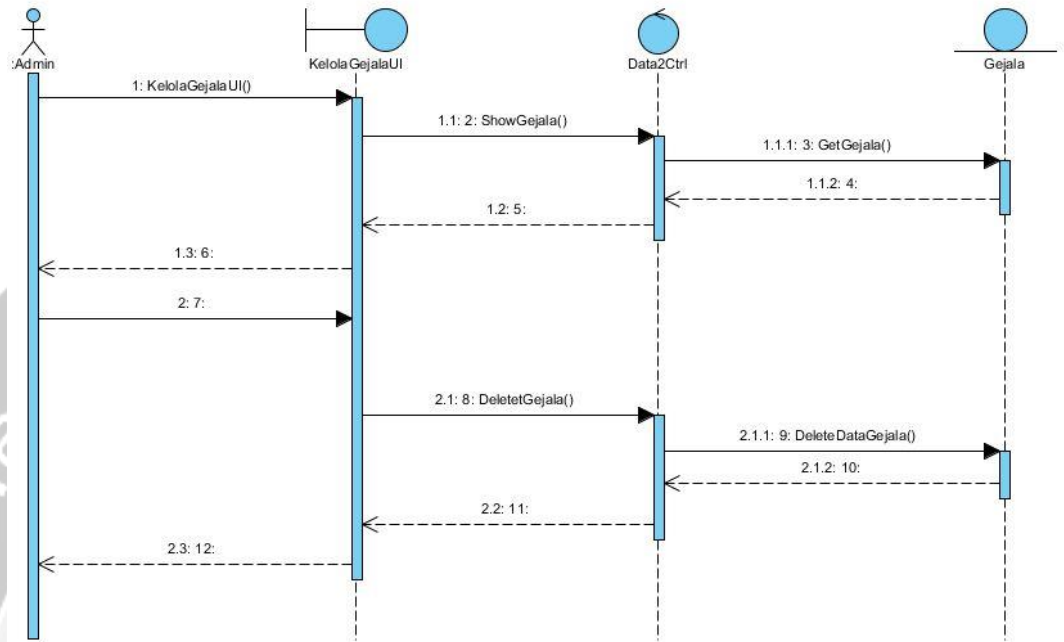
2.2.1.10. Fungsi Tambah Data Gejala



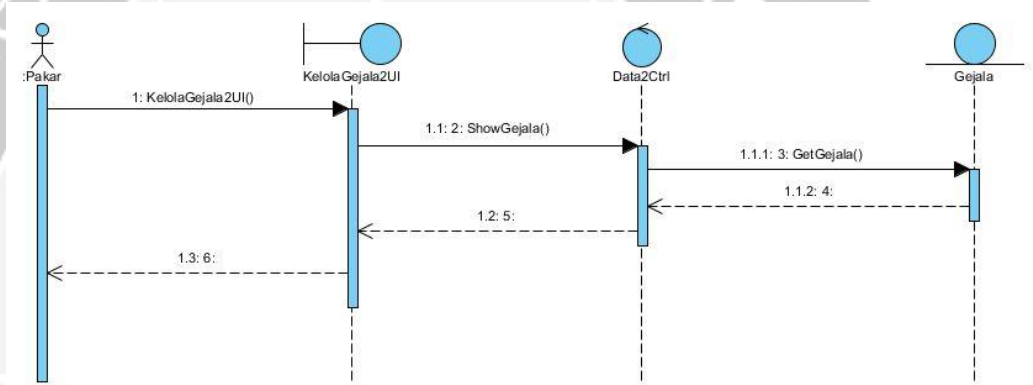
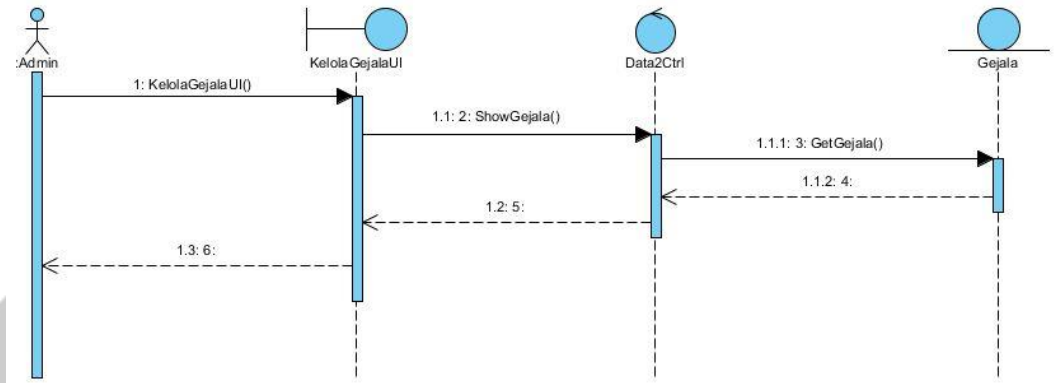
2.2.1.11. Fungsi Ubah Data Gejala



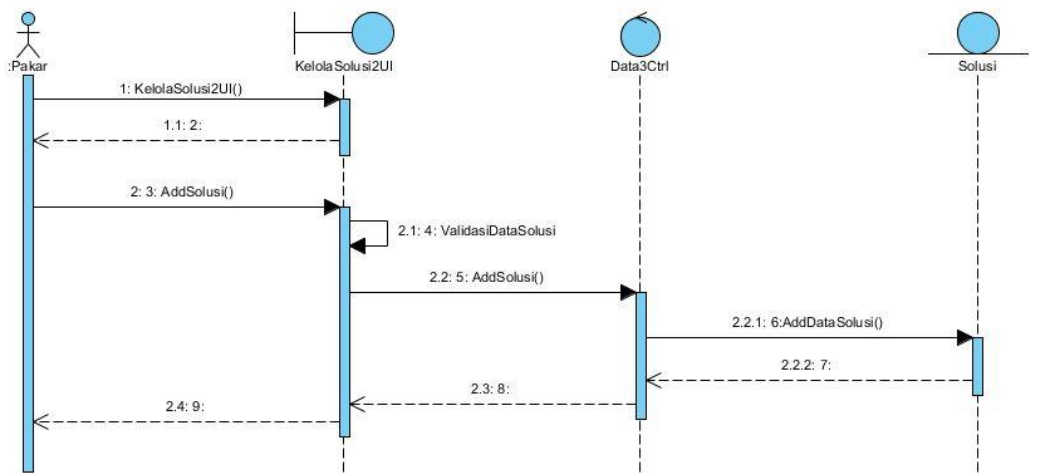
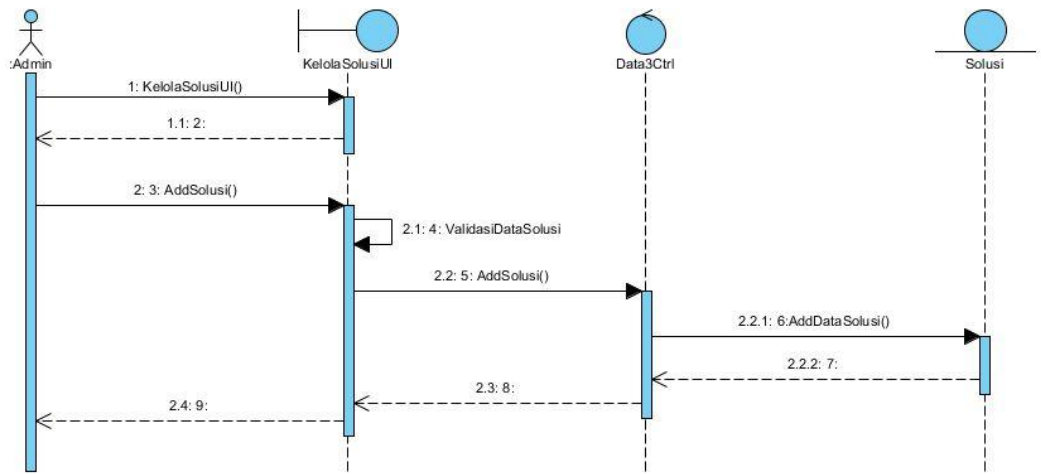
2.2.1.12. Fungsi Hapus Data Gejala



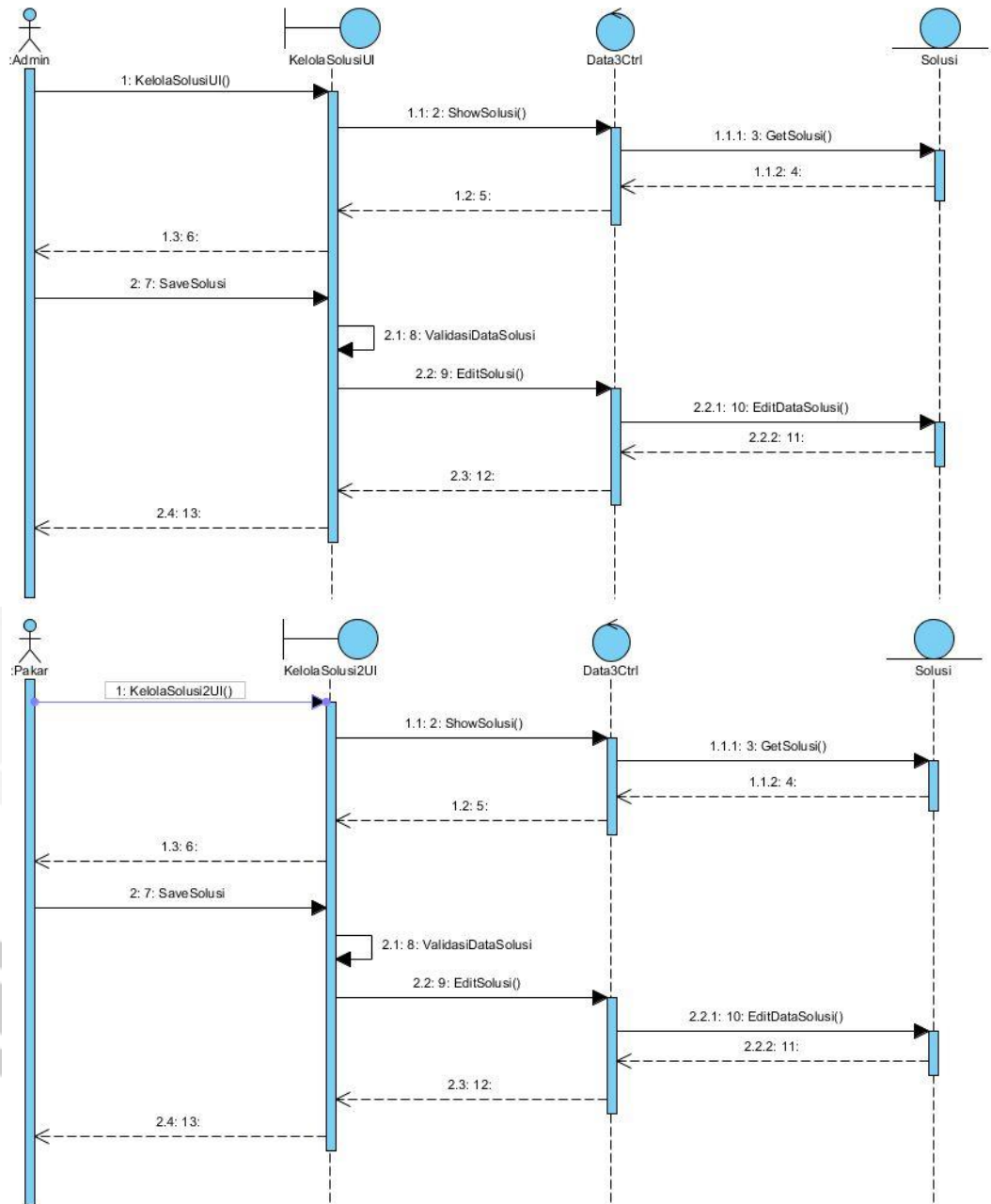
2.2.1.13. Fungsi Tampil Data Gejala



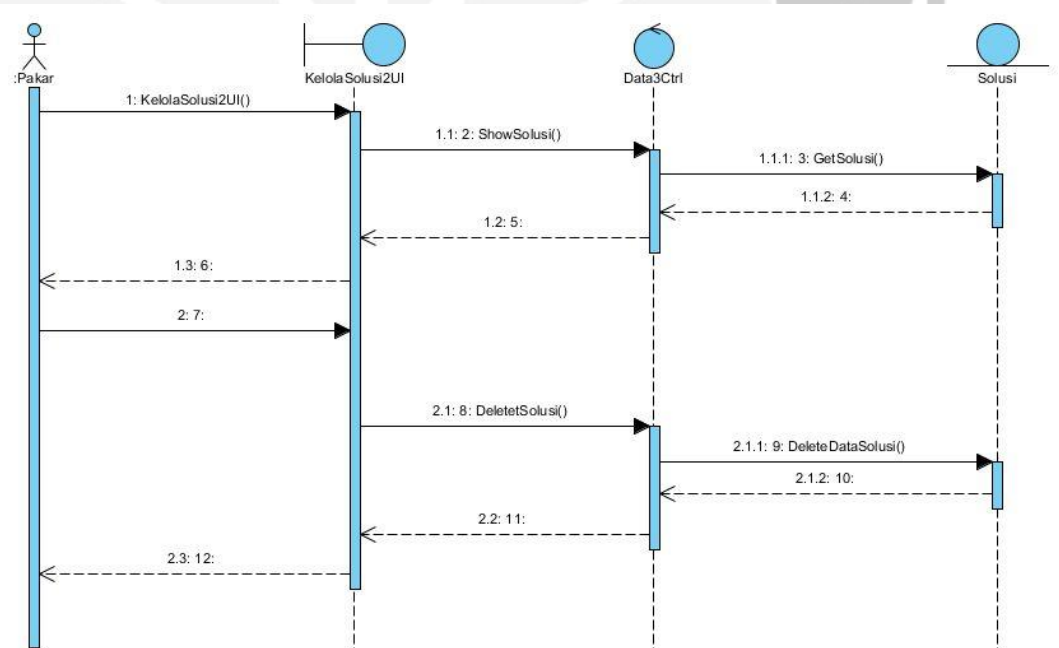
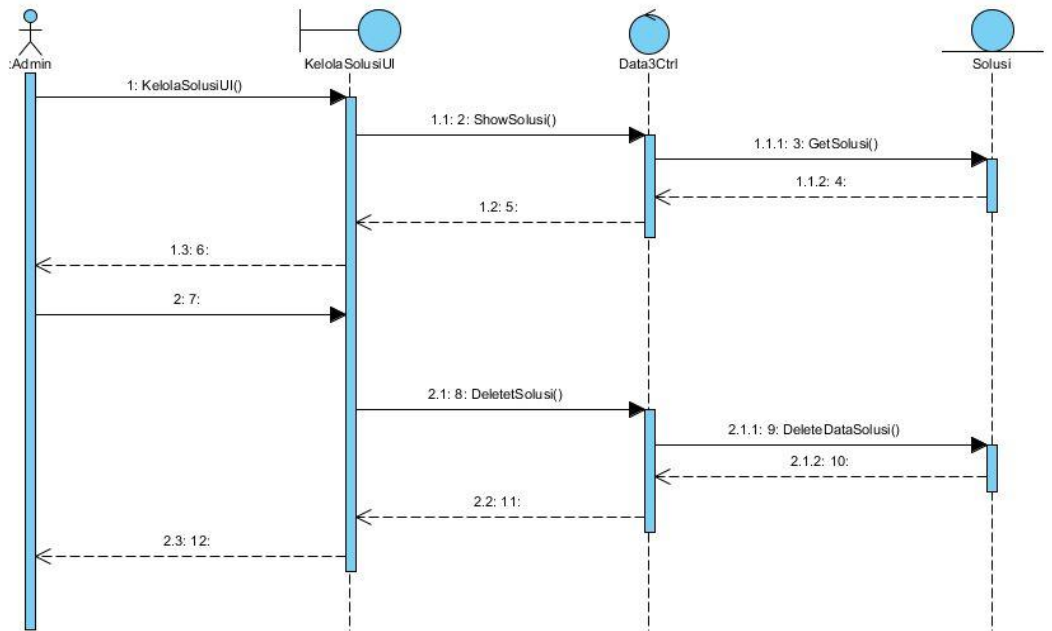
2.2.1.14. Fungsi Tambah Data Solusi



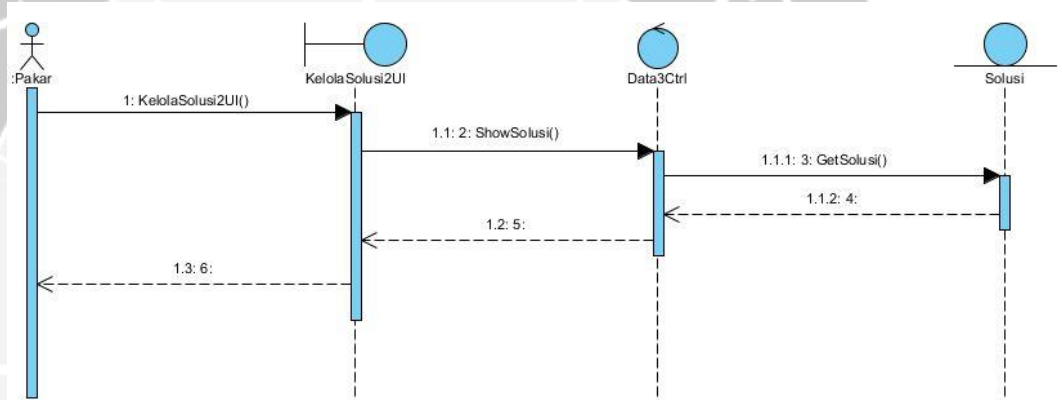
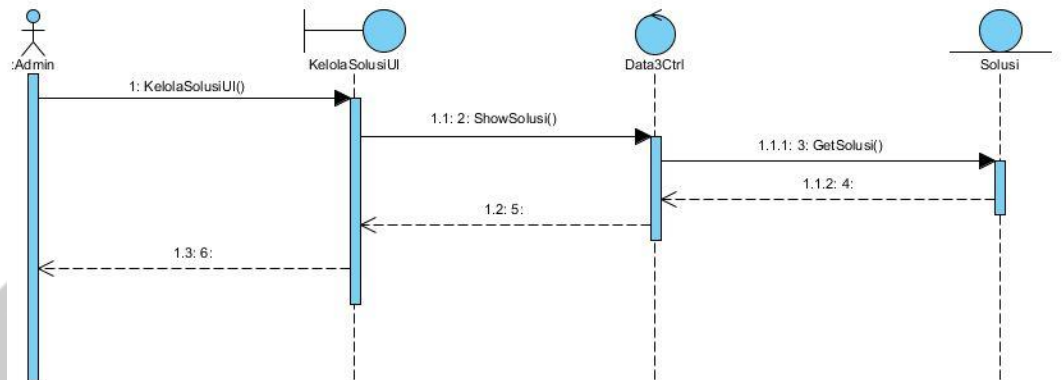
2.2.1.15. Fungsi Ubah Data Solusi



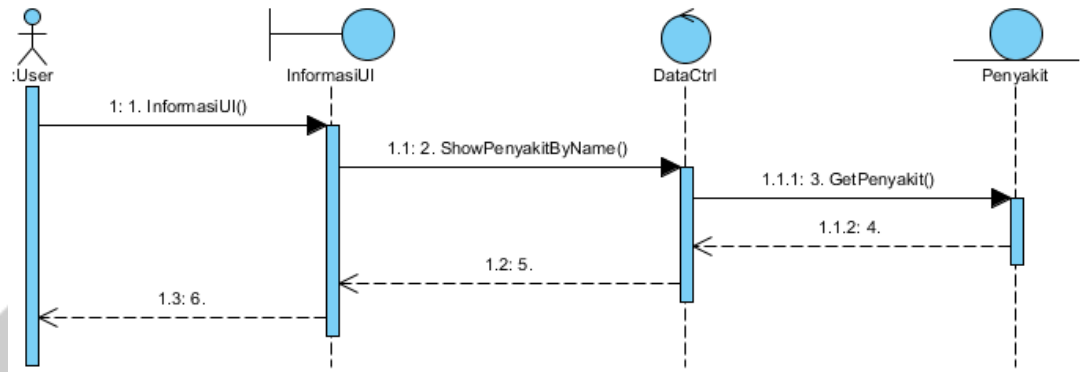
2.2.1.16. Fungsi Hapus Data Solusi



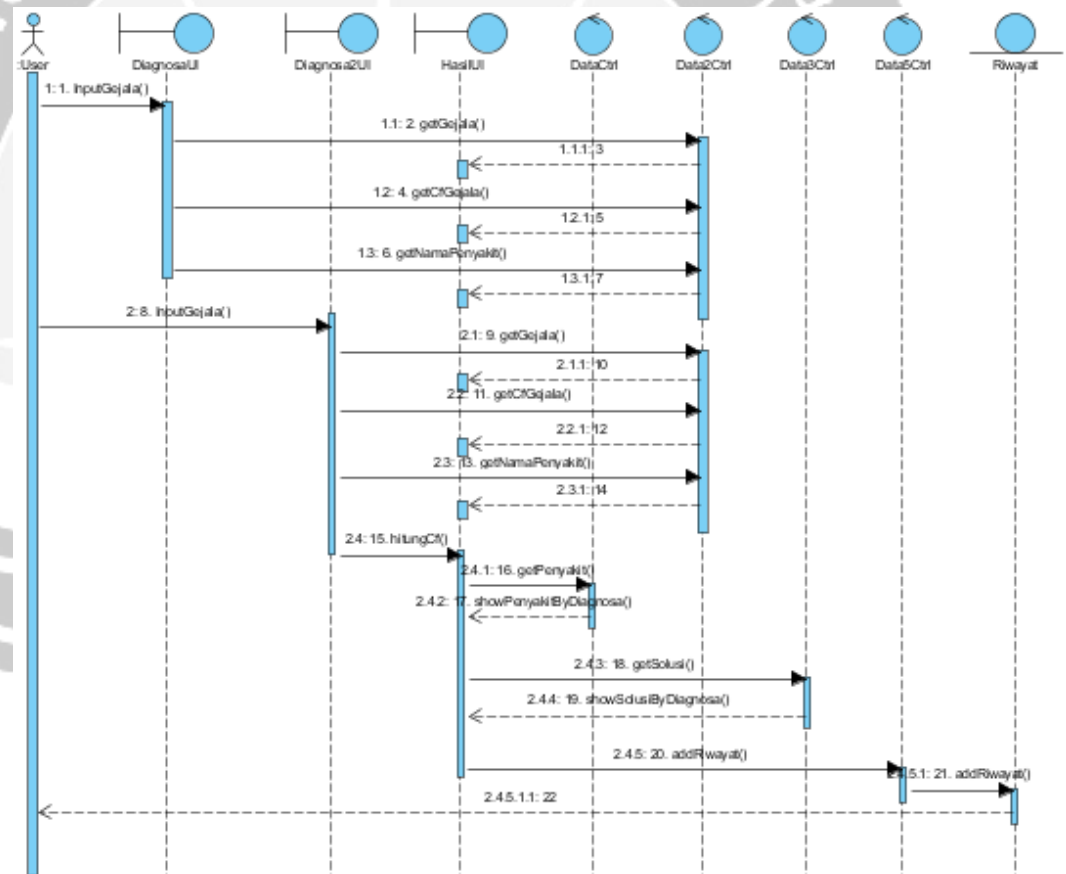
2.2.1.17. Fungsi Tampil Data Solusi



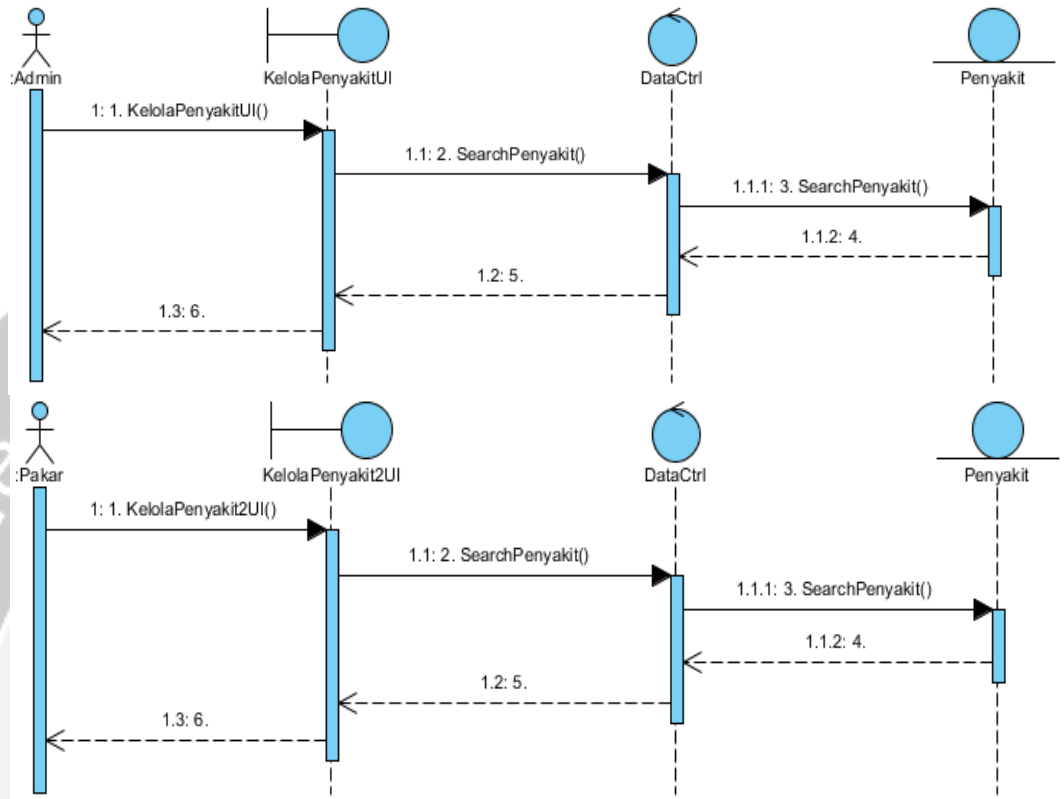
2.2.1.18. Fungsi Informasi



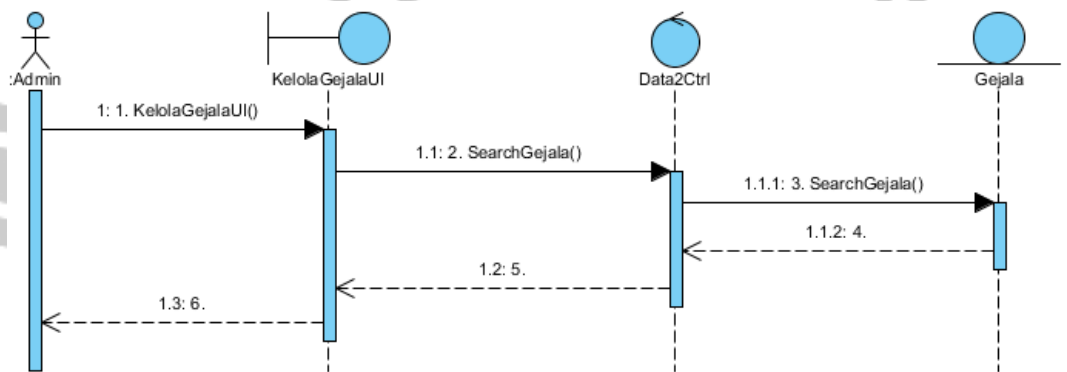
2.2.1.19. Fungsi Diagnosa

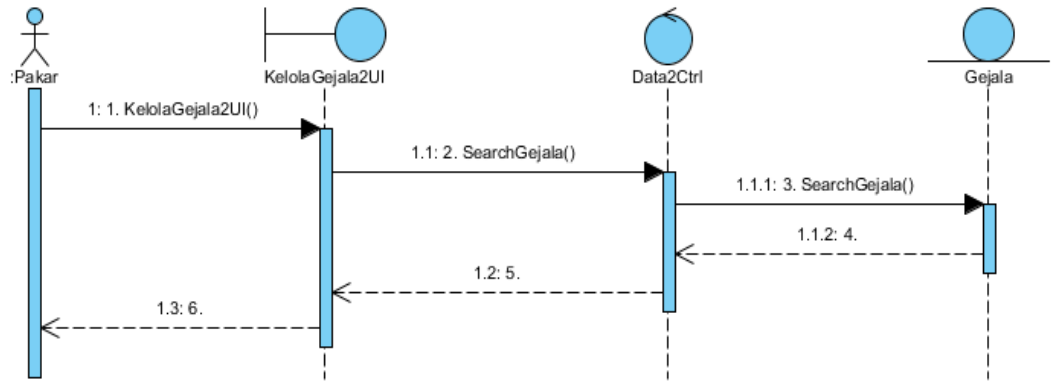


2.2.1.20. Fungsi Cari Data Penyakit

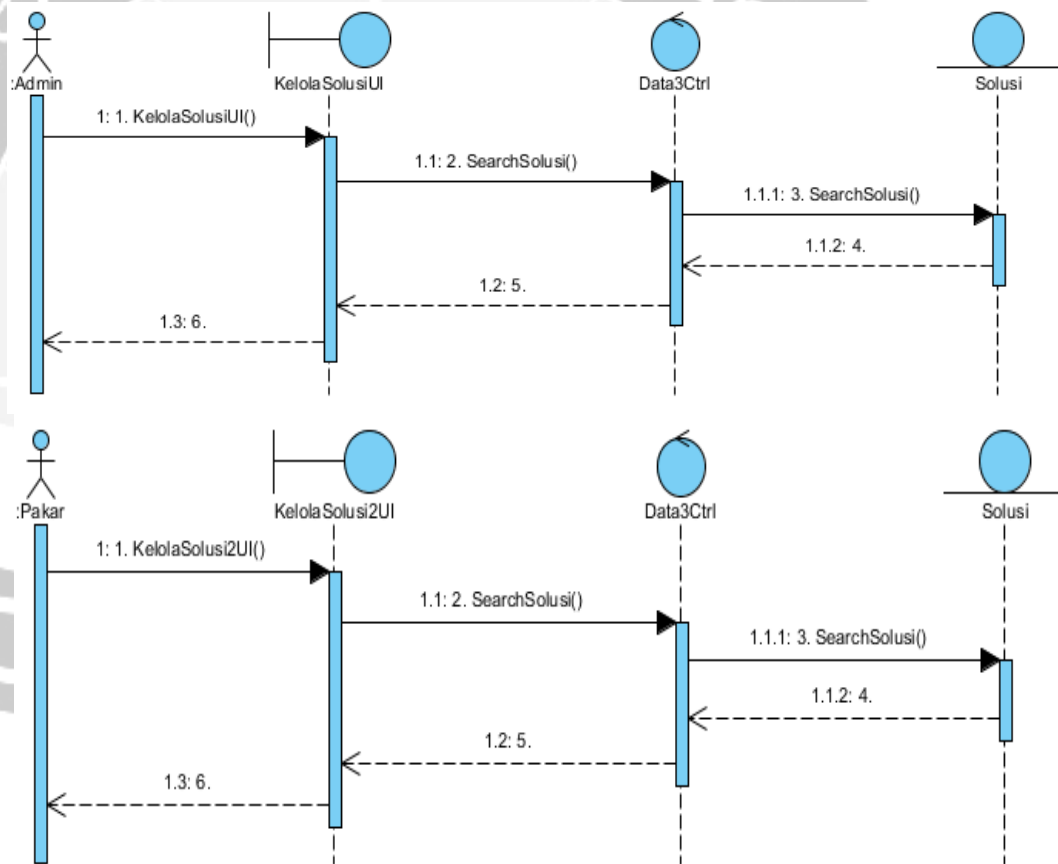


2.2.1.21. Fungsi Cari Data Gejala

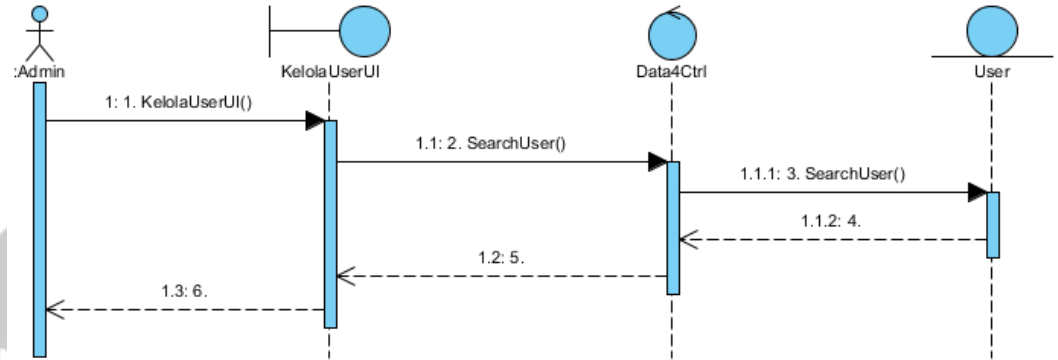




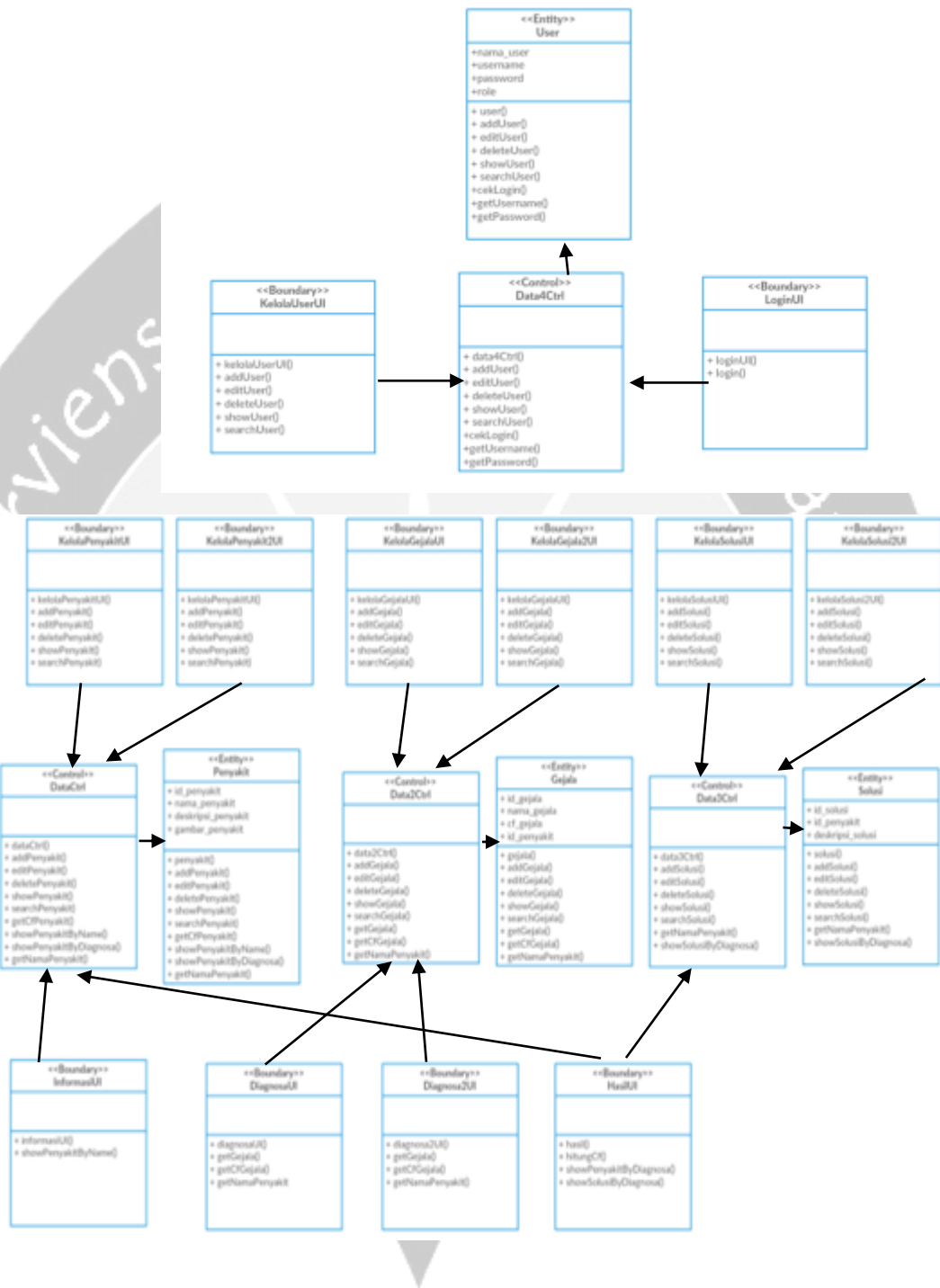
2.2.1.22. Fungsi Cari Data Solusi

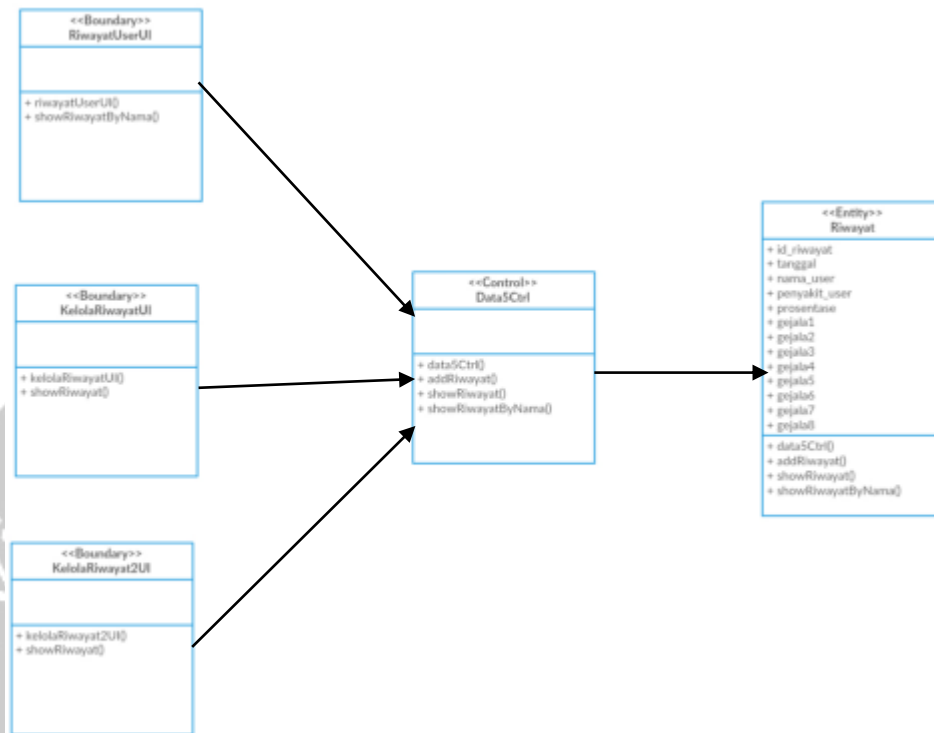


2.2.1.23. Fungsi Cari Data User



2.2.2. Class Diagram





2.2.3. Class Diagram Specific Descriptions

2.2.3.1. Specific Design Class User

User	<<entity>>
+nama_user string, digunakan untuk menyimpan data nama pengguna +username string, digunakan untuk menyimpan data username +password string, digunakan untuk menyimpan data password +role string, digunakan untuk menyimpan data role	
+user() Default konstruktor, digunakan untuk inialisasi semua attribute dari kelas ini.	

+addUser() digunakan untuk mengambil data user dari form yang diinputkan oleh admin.

+editUser() digunakan untuk mengedit data pengguna yang ada di basis data

+deleteUser() digunakan untuk menghapus data pengguna yang ada di database

+showUser() digunakan untuk menampilkan semua data pengguna yang ada di database

+searchUser() digunakan untuk mencari data pengguna yang ada di database

+cekLogin() digunakan untuk mengecek data username dan password yang ada di database

+getUsername() digunakan untuk mendapatkan data username yang ada di database

+getPassword() digunakan untuk mendapatkan password yang ada di database

2.2.3.2. Specific Design Class KelolaUserUI

KelolaUserUI	<<boundary>>
<pre>+kelolaUserUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</pre>	

+addUser() digunakan untuk mengambil data user dari form yang diinputkan oleh admin.

+editUser() digunakan untuk mengedit data pengguna yang ada di basis data

+deleteUser() digunakan untuk menghapus data pengguna yang ada di database

+showUser() digunakan untuk menampilkan semua data pengguna yang ada di database

+searchUser() digunakan untuk mencari data pengguna yang ada di database

2.2.3.3. Specific Design Class Data4Ctrl

Data4Ctrl	<<control>>
<p>+data4Ctrl()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addUser() digunakan untuk mengambil data user dari form yang diinputkan oleh admin.</p> <p>+editUser() digunakan untuk mengedit data pengguna yang ada di basis data</p> <p>+deleteUser() digunakan untuk menghapus data pengguna yang ada di database</p> <p>+showUser() digunakan untuk menampilkan semua data pengguna yang ada di database</p> <p>+searchUser() digunakan untuk mencari data pengguna yang ada di database</p>	

+cekLogin() digunakan untuk mengecek data username dan password yang ada di database

+getUsername() digunakan untuk mendapatkan data username yang ada di database

+getPassword() digunakan untuk mendapatkan password yang ada di database

2.2.3.4. Specific Design Class LoginUI

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

2.2.3.5. Specific Design Class KelolaPenyakitUI

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

+addPenyakit() digunakan untuk mengambil data penyakit dari form yang diinputkan oleh admin.

+editPenyakit() digunakan untuk mengedit data penyakit yang ada di basis data

+deletePenyakit() digunakan untuk menghapus data penyakit yang ada di database

+showPenyakit() digunakan untuk menampilkan semua data penyakit yang ada di database

+searchPenyakit() digunakan untuk mencari data penyakit yang ada di database

2.2.3.6. Specific Design Class KelolaPenyakit2UI

KelolaPenyakit2UI	<<boundary>>
<pre>+kelolaPenyakit2UI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +addPenyakit() digunakan untuk mengambil data penyakit dari form yang diinputkan oleh pakar. +editPenyakit() digunakan untuk mengedit data penyakit yang ada di basis data +deletePenyakit() digunakan untuk menghapus data penyakit yang ada di database</pre>	

+showPenyakit() digunakan untuk menampilkan semua data penyakit yang ada di database

+searchPenyakit() digunakan untuk mencari data penyakit yang ada di database

2.2.3.7. Specific Design Class DataCtrl

DataCtrl	<<control>>
<p>+dataCtrl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addPenyakit() digunakan untuk mengambil data penyakit dari form yang diinputkan oleh pengguna.</p> <p>+editPenyakit() digunakan untuk mengedit data penyakit yang ada di basis data</p> <p>+deletePenyakit() digunakan untuk menghapus data penyakit yang ada di database</p> <p>+showPenyakit() digunakan untuk menampilkan semua data penyakit yang ada di database</p> <p>+searchPenyakit() digunakan untuk mencari data penyakit yang ada di database</p>	

+getCfPenyakit() digunakan untuk mendapatkan nilai kepastian dari suatu penyakit

+showPenyakitByName() digunakan untuk menampilkan data penyakit berdasarkan nama penyakit yang ada di database

+showPenyakitByDiagnosa() digunakan untuk menampilkan data penyakit berdasarkan diagnosa yang ada di database

+getNamaPenyakit() digunakan untuk mendapatkan nama penyakit saat diagnosa

2.2.3.8. Specific Design Class Penyakit

Penyakit	<<entity>>
<p>+id_penyakit integer, digunakan untuk menyimpan data id penyakit</p> <p>+nama_penyakit string, digunakan untuk menyimpan data nama penyakit</p> <p>+deskripsi_penyakit string, digunakan untuk menyimpan data deskripsi penyakit</p> <p>+gambar_penyakit string, digunakan untuk menyimpan data alamat gambar penyakit</p>	
<p>+penyakit()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p>	

+addPenyakit() digunakan untuk mengambil data penyakit dari form yang diinputkan oleh pengguna.

+editPenyakit() digunakan untuk mengedit data penyakit yang ada di basis data

+deletePenyakit() digunakan untuk menghapus data penyakit yang ada di database

+showPenyakit() digunakan untuk menampilkan semua data penyakit yang ada di database

+searchPenyakit() digunakan untuk mencari data penyakit yang ada di database

+getCfPenyakit() digunakan untuk mendapatkan nilai kepastian dari suatu penyakit

+showPenyakitByName() digunakan untuk menampilkan data penyakit berdasarkan nama penyakit yang ada di database

+showPenyakitByDiagnosa() digunakan untuk menampilkan data penyakit berdasarkan diagnosa yang ada di database

+getNamaPenyakit() digunakan untuk mendapatkan nama penyakit saat diagnosa

2.2.3.9. Specific Design Class KelolaGejalaUI

KelolaGejalaUI	<<boundary>>

+kelolaGejalaUI()

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

+addGejala() digunakan untuk mengambil data gejala dari form yang diinputkan oleh admin.

+editGejala() digunakan untuk mengedit data gejala yang ada di basis data

+deleteGejala() digunakan untuk menghapus data gejala yang ada di database

+showGejala() digunakan untuk menampilkan semua data gejala yang ada di database

+searchGejala() digunakan untuk mencari data gejala yang ada di database

2.2.3.10. Specific Design Class KelolaGejala2UI

KelolaGejala2UI	<<boundary>>
<p>+kelolaGejala2UI()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addGejala() digunakan untuk mengambil data gejala dari form yang diinputkan oleh pakar.</p> <p>+editGejala() digunakan untuk mengedit data gejala yang ada di basis data</p>	

+deleteGejala() digunakan untuk menghapus data gejala yang ada di database

+showGejala() digunakan untuk menampilkan semua data gejala yang ada di database

+searchGejala() digunakan untuk mencari data gejala yang ada di database

2.2.3.11. Specific Design Class Data2Ctrl

Data2Ctrl	<<control>>
<p>+data2Ctrl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addGejala()digunakan untuk mengambil data gejala dari form yang diinputkan oleh pengguna.</p> <p>+editGejala() digunakan untuk mengedit data gejala yang ada di basis data</p> <p>+deleteGejala() digunakan untuk menghapus data gejala yang ada di database</p> <p>+showGejala() digunakan untuk menampilkan semua data gejala yang ada di database</p> <p>+searchGejala() digunakan untuk mencari data gejala yang ada di database</p>	

+getGejala() digunakan untuk mendapatkan data gejala saat proses diagnosa

+getCfGejala() digunakan untuk mendapatkan nilai kepercayaan berdasarkan data gejala saat proses diagnosa

+getNamaPenyakit() digunakan untuk mendapatkan nama penyakit pada saat gejala dimasukkan ke dalam database

2.2.3.12. Specific Design Class Gejala

Gejala	<<entity>>
<p>+id_gejala integer, digunakan untuk menyimpan data id gejala</p> <p>+nama_gejala string, digunakan untuk menyimpan data nama gejala</p> <p>+cf_gejala double, digunakan untuk menyimpan data nilai kepercayaan gejala</p> <p>+id_gejala integer, digunakan untuk menyimpan data id penyakit</p>	
<p>+gejala() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addGejala() digunakan untuk mengambil data gejala dari form yang diinputkan oleh admin.</p> <p>+editGejala() digunakan untuk mengedit data gejala yang ada di basis data</p>	

+deleteGejala() digunakan untuk menghapus data gejala yang ada di database

+showGejala() digunakan untuk menampilkan semua data gejala yang ada di database

+searchGejala() digunakan untuk mencari data gejala yang ada di database

+getGejala() digunakan untuk mendapatkan data gejala saat proses diagnosa

+getCfGejala() digunakan untuk mendapatkan nilai kepercayaan berdasarkan data gejala saat proses diagnosa

+getNamaPenyakit() digunakan untuk mendapatkan nama penyakit pada saat gejala dimasukkan ke dalam database

2.2.3.13. Specific Design Class KelolaSolusiUI

KelolaSolusiUI	<<boundary>>
<p>+kelolaSolusiUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addSolusi() digunakan untuk mengambil data gejala dari form yang diinputkan oleh admin.</p> <p>+editSolusi() digunakan untuk mengedit data solusi yang ada di basis data</p>	

+deleteSolusi() digunakan untuk menghapus data solusi yang ada di database

+showSolusi() digunakan untuk menampilkan semua data solusi yang ada di database

+searchSolusi() digunakan untuk mencari data solusi yang ada di database

2.2.3.14. Specific Design Class KelolaSolusi2UI

KelolaSolusi2UI	<<boundary>>
<p>+kelolaSolusi2UI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addSolusi() digunakan untuk mengambil data gejala dari form yang diinputkan oleh pakar.</p> <p>+editSolusi() digunakan untuk mengedit data solusi yang ada di basis data</p> <p>+deleteSolusi() digunakan untuk menghapus data solusi yang ada di database</p> <p>+showSolusi() digunakan untuk menampilkan semua data solusi yang ada di database</p> <p>+searchSolusi() digunakan untuk mencari data solusi yang ada di database</p>	

2.2.3.15. Specific Design Class Data3Ctrl

Data3Ctrl	<<control>>
<pre> +data3Ctrl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +addSolusi() digunakan untuk mengambil data gejala dari form yang diinputkan oleh admin. +editSolusi() digunakan untuk mengedit data solusi yang ada di basis data +deleteSolusi() digunakan untuk menghapus data solusi yang ada di database +showSolusi() digunakan untuk menampilkan semua data solusi yang ada di database +searchSolusi() digunakan untuk mencari data solusi yang ada di database +getNamaPenyakit() digunakan untuk mendapatkan nama penyakit berdasarkan data solusi yang ada di database +showSolusiByDiagnosa() digunakan untuk mendapatkan data solusi yang ada di database yang digunakan pada saat proses diagnosa </pre>	

2.2.3.16. Specific Design Class Solusi

Solusi	<<entity>>
<p>+id_solusi integer, digunakan untuk menyimpan data id solusi</p> <p>+id_penyakit integer, digunakan untuk menyimpan data id penyakit</p> <p>+deskripsi_solusi string, digunakan untuk menyimpan data deskripsi solusi</p>	
<p>+solusi()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addSolusi() digunakan untuk mengambil data gejala dari form yang diinputkan oleh pengguna.</p> <p>+editSolusi() digunakan untuk mengedit data solusi yang ada di basis data</p> <p>+deleteSolusi() digunakan untuk menghapus data solusi yang ada di database</p> <p>+showSolusi() digunakan untuk menampilkan semua data solusi yang ada di database</p> <p>+searchSolusi() digunakan untuk mencari data solusi yang ada di database</p> <p>+getNamaPenyakit() digunakan untuk mendapatkan nama penyakit berdasarkan data solusi yang ada di database</p> <p>+showSolusiByDiagnosa() digunakan untuk mendapatkan data solusi yang ada di</p>	

database yang digunakan pada saat proses diagnosa

2.2.3.17. Specific Design Class DiagnosaUI

DiagnosaUI	<<boundary>>
<pre>+diagnosaUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getGejala(), digunakan untuk mendapatkan data gejala yang dipilih oleh pengguna +getCfGejala(), digunakan untuk mendapatkan nilai kepercayaan gejala berdasarkan gejala yang dipilih oleh pengguna +getNamaPenyakit(), digunakan untuk mendapatkan nama penyakit berdasarkan gejala yang dipilih oleh pengguna</pre>	

2.2.3.18. Specific Design Class Diagnosa2UI

Diagnosa2UI	<<boundary>>
<pre>+diagnosa2UI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</pre>	

+getGejala(), digunakan untuk mendapatkan data gejala yang dipilih oleh pengguna

+getCfGejala(), digunakan untuk mendapatkan nilai kepercayaan gejala berdasarkan gejala yang dipilih oleh pengguna

+getNamaPenyakit(), digunakan untuk mendapatkan nama penyakit berdasarkan gejala yang dipilih oleh pengguna

2.2.3.19. Specific Design Class HasilUI

HasilUI	<<boundary>>
<p>+hasilUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+hitungCf(), digunakan untuk menghitung nilai kepercayaan berdasarkan gejala yang telah dipilih oleh pengguna</p> <p>+showPenyakitByDiagnosa(), digunakan untuk menampilkan data penyakit berdasarkan hasil diagnosa</p> <p>+showSolusiByDiagnosa(), digunakan untuk menampilkan data solusi berdasarkan hasil diagnosa</p>	

2.2.3.20. Specific Design Class RiwayatUserUI

RiwayatUserUI	<<boundary>>
----------------------	---------------------------------

<pre>+riwayatUserUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <pre>+showRiwayatByNama(),</pre> <p>digunakan untuk menampilkan data riwayat diagnosa berdasarkan nama pengguna</p>

2.2.3.21. Specific Design Class KelolaRiwayatUI

KelolaRiwayatUI	<<boundary>>
<pre>+kelolaRiwayatUI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <pre>+showRiwayat(),</pre> <p>digunakan untuk menampilkan data riwayat yang tersimpan di database</p>	

2.2.3.22. Specific Design Class Kelola riwayat2UI

KelolaRiwayat2UI	<<boundary>>
<pre>+kelolaRiwayat2UI()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p>	

+showRiwayat(), digunakan untuk menampilkan data riwayat yang tersimpan di database

2.2.3.23. Specific Design Class Data5Ctrl

Data5Ctrl	<<control>>
<p>+data5Ctrl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+addRiwayat(), digunakan untuk menambahkan data riwayat ke dalam database setelah pengguna setiap melakukan diagnosa</p> <p>+showRiwayatByNama(), digunakan untuk menampilkan data riwayat diagnosa berdasarkan nama pengguna</p> <p>+showRiwayat(), digunakan untuk menampilkan data riwayat yang tersimpan di database</p>	

2.2.3.24. Specific Design Class Riwayat

Riwayat	<<entity>>
<p>+id_riwayat integer, digunakan untuk menyimpan data id riwayat</p> <p>+tanggal date, digunakan untuk menyimpan data tanggal riwayat</p> <p>+nama_user string, digunakan untuk menyimpan data nama pengguna</p>	

+penyakit_user string, digunakan untuk menyimpan data penyakit pengguna

+prosentase double, digunakan untuk menyimpan data prosentase penyakit pengguna

+gejala1 string, digunakan untuk menyimpan data gejala pertama

+gejala2 string, digunakan untuk menyimpan data gejala kedua

+gejala3 string, digunakan untuk menyimpan data gejala ketiga

+gejala4 string, digunakan untuk menyimpan data gejala keempat

+gejala5 string, digunakan untuk menyimpan data gejala kelima

+gejala6 string, digunakan untuk menyimpan data gejala keenam

+gejala7 string, digunakan untuk menyimpan data gejala ketujuh

+gejala8 string, digunakan untuk menyimpan data gejala kedelapan

+riwayat()

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

+addRiwayat(), digunakan untuk menambahkan data riwayat ke dalam database setelah pengguna setiap melakukan diagnosa

+showRiwayatByNama(), digunakan untuk menampilkan data riwayat diagnosa berdasarkan nama pengguna

+showRiwayat(), digunakan untuk menampilkan data riwayat yang tersimpan di database

2.2.3.25. Specific Design Class Informasi

Informasi	<<boundary>>
<p>+informasi() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+showPenyakitByName() digunakan untuk menampilkan data penyakit berdasarkan nama penyakit yang ada di database</p>	

3. PERANCANGAN DATA

3.1. Dekomposisi Data

4.3.1.1. Deskripsi Entitas TBL_kat1

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 1, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 1
CF	Double		Nilai kepercayaan dari kategori 1

ID_PENYAKIT	Integer	Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit
-------------	---------	----------------------------------------------------------------------

3.1.2. Deskripsi Entitas TBL_kat2

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 2, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 2
CF	Double		Nilai kepercayaan dari kategori 2
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.3. Deskripsi Entitas TBL_kat3

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 3, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 3

CF	Double	Nilai kepercayaan dari kategori 3
ID_PENYAKIT	Integer	Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.4. Deskripsi Entitas TBL_kat4

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 4, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 4
CF	Double		Nilai kepercayaan dari kategori 4
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.5. Deskripsi Entitas TBL_kat5

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 5, Primary Key

GEJALA	Variable Characters	1000	Nama gejala dari kategori 5
CF	Double		Nilai kepercayaan dari kategori 5
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.6. Deskripsi Entitas TBL_kat6

Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 6, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 6
CF	Double		Nilai kepercayaan dari kategori 6
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.7. Deskripsi Entitas TBL_kat7			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 7, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 7
CF	Double		Nilai kepercayaan dari kategori 7
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.8. Deskripsi Entitas TBL_kat8			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari kategori 8, Primary Key
GEJALA	Variable Characters	1000	Nama gejala dari kategori 8
CF	Double		Nilai kepercayaan dari kategori 8
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit

3.1.9. Deskripsi Entitas TBL_penyakit			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari penyakit, Primary Key
NAMA	Variable Characters	1000	Nama dari penyakit
DESKRIPSI	Variable Characters	1000	Deskripsi dari penyakit
GAMBAR	Text		Gambar dari penyakit

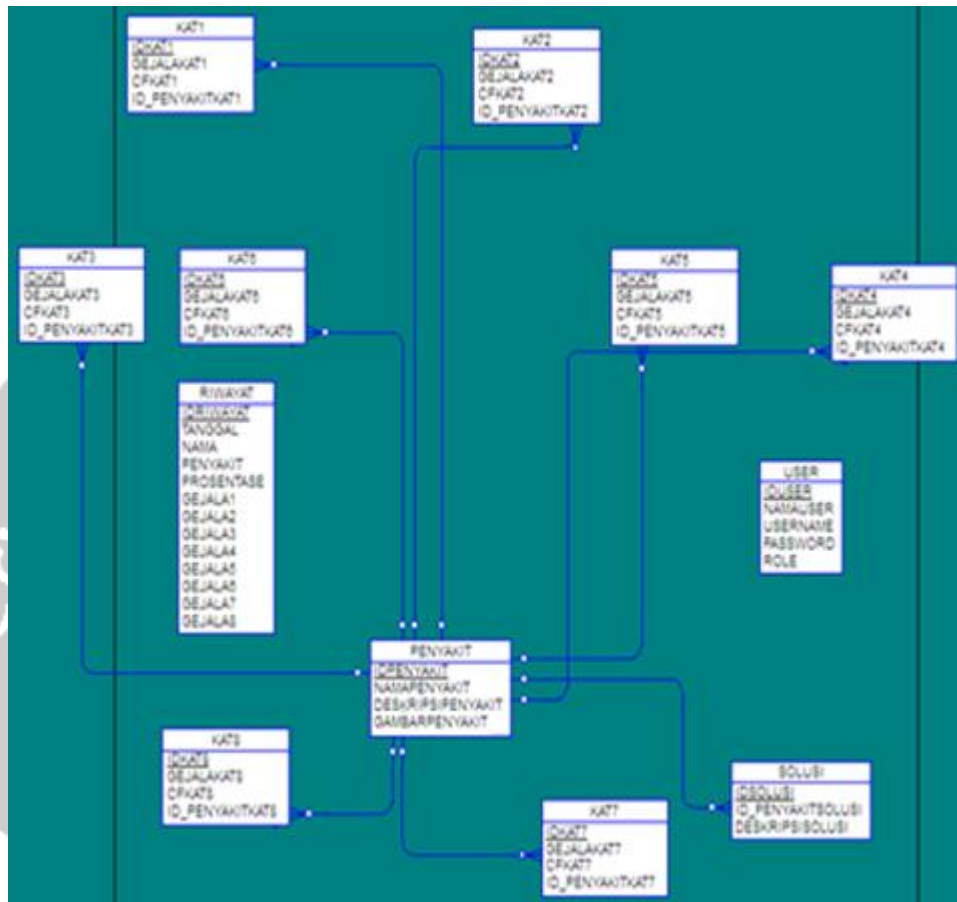
3.1.10. Deskripsi Entitas TBL_solusi			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari solusi, Primary Key
ID_PENYAKIT	Integer		Id dari penyakit, foreign key terhadap ID_PENYAKIT pada TBL_penyakit
SOLUSI	Variable Characters	1000	Deskripsi dari solusi

3.1.11. Deskripsi Entitas TBL_user			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari pengguna, Primary Key
NAMA	Variable Characters	1000	Nama asli dari pengguna
USERNAME	Variable Characters	1000	Username dari pengguna
PASSWORD	Variable Characters	1000	Password dari pengguna
ROLE	Variable Characters	1000	Role dari pengguna

3.1.12. Deskripsi Entitas TBL_riwayat			
Nama	Tipe	Panjang	Keterangan
ID	Integer		Id dari riwayat, Primary Key
TANGGAL	Date		Tanggal Diagnosa
NAMA	Variable Characters	1000	Nama asli dari pengguna
PENYAKIT	Variable Characters	1000	Nama penyakit hasil diagnosa dari pengguna
PROSENTASE	Double	1000	Prosentase hasil diagnosa dari pengguna

GEJALA1	Variable Characters	1000	Gejala pertama yang dipilih oleh pengguna
GEJALA2	Variable Characters	1000	Gejala kedua yang dipilih oleh pengguna
GEJALA3	Variable Characters	1000	Gejala ketiga yang dipilih oleh pengguna
GEJALA4	Variable Characters	1000	Gejala keempat yang dipilih oleh pengguna
GEJALA5	Variable Characters	1000	Gejala kelima yang dipilih oleh pengguna
GEJALA6	Variable Characters	1000	Gejala keenam yang dipilih oleh pengguna
GEJALA7	Variable Characters	1000	Gejala ketujuh yang dipilih oleh pengguna
GEJALA8	Variable Characters	1000	Gejala kedelapan yang dipilih oleh pengguna

3.2. Physical Data Model



4. Deskripsi Perancangan Antar Muka

4.1. Antarmuka Halaman Login



Antarmuka tersebut dapat digunakan oleh admin, pakar, dan user masuk ke dalam SiPaPa. Masuk ke dalam SiPaPa, pengguna diminta untuk memasukkan username dan password, kemudian tekan tombol masuk untuk masuk ke dalam SiPaPa. Tombol batal digunakan untuk batal masuk ke dalam SiPaPa. Pada saat pengguna menekan tombol masuk, sistem akan mengecek apakah username dan password sesuai yang tersimpan pada database, jika username dan password tidak sesuai, maka sistem akan memberikan peringatan username atau password tidak sesuai.

4.2. Antarmuka Halaman Menu Admin



Antarmuka tersebut merupakan menu yang dapat digunakan oleh admin untuk melakukan pengelolaan data yang terdapat di SiPaPa. Admin dapat memilih menu yang telah disediakan, ketika admin memilih suatu menu, maka menu tersebut menuju antarmuka lain berdasarkan menu yang dipilih oleh admin.

4.3. Antarmuka Halaman Menu Pakar



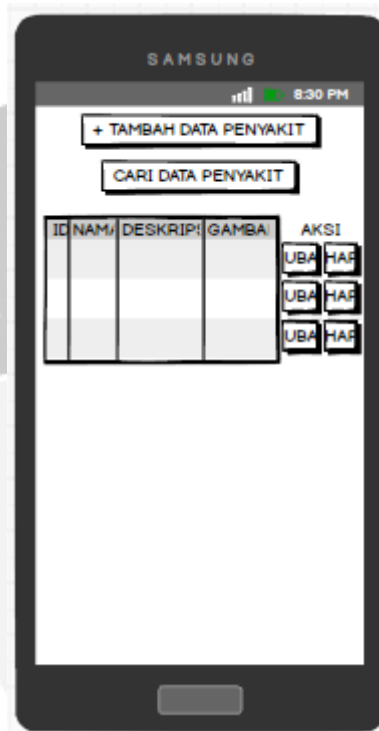
Antarmuka tersebut merupakan menu yang dapat digunakan oleh pakar untuk melakukan pengelolaan data yang terdapat di SiPaPa. Pakar dapat memilih menu yang telah disediakan, ketika pakar memilih suatu menu, maka menu tersebut menuju antarmuka lain berdasarkan menu yang dipilih oleh pakar.

4.4. Antarmuka Halaman Menu User



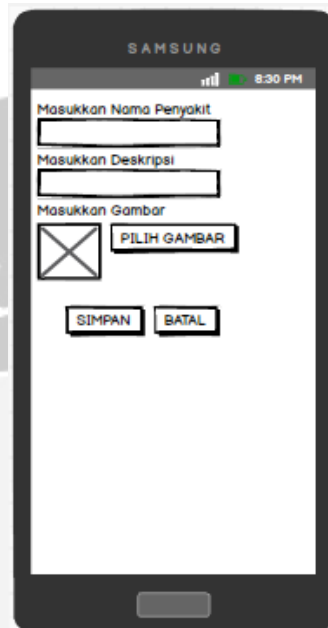
Antarmuka tersebut merupakan menu yang dapat digunakan oleh *user* untuk melakukan diagnosa, mengubah password, melihat informasi mengenai penyakit pada organ pankreas, dan melihat riwayat mengenai diagnosa sebelumnya berdasarkan *user* tersebut. *User* dapat memilih menu yang telah disediakan, ketika *user* memilih suatu menu, maka menu tersebut menuju antarmuka lain berdasarkan menu yang dipilih oleh *user*.

4.5. Antarmuka Halaman Kelola Data Penyakit



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mengelola data penyakit. Admin dan pakar dapat melihat data penyakit yang tersimpan pada database, mencari data penyakit, menambahkan data penyakit, mengubah data penyakit, dan menghapus data penyakit.

4.6. Antarmuka Halaman Tambah Ubah Data Penyakit



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk menambahkan atau mengubah data penyakit yang tersimpan di dalam database. Pada saat admin dan pakar ingin menambahkan atau mengubah data penyakit, admin dan pakar diminta untuk memasukkan nama penyakit, deskripsi penyakit, dan memasukkan gambar penyakit. Setelah admin dan pakar selesai memasukkan data yang dibutuhkan, maka admin dan pakar diminta untuk menekan tombol simpan. Pada saat proses tambah data penyakit, maka sistem akan mengecek apakah nama penyakit yang dimasukkan sudah ada di dalam database atau tidak, jika sudah ada di dalam database maka sistem memberi peringatan bahwa nama penyakit sudah ada, jika belum ada maka data yang dimasukkan tadi disimpan di dalam database. Pada saat proses ubah data, maka data yang dimasukkan tadi akan disimpan di dalam

database. Tombol batal digunakan untuk kembali ke menu kelola data penyakit.

4.7. Antarmuka Halaman Kelola Data Gejala



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mengelola data gejala penyakit. Pada menu kelola data gejala, admin dan pakar dapat menambahkan data gejala yang nantinya akan disimpan di dalam setiap kategori. Pada menu kelola data gejala, admin dan pakar dapat menuju setiap menu kategori gejala penyakit yang telah disediakan. Pada setiap menu kategori, admin dan pakar dapat melihat gejala yang telah disediakan, menghapus data gejala, mencari data gejala, dan mengubah data setiap kategori gejala penyakit.

4.8. Antarmuka Halaman Tambah Data Gejala



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk menambahkan data gejala. Pada saat proses tambah data gejala, admin dan pakar diminta untuk memasukkan nama gejala, nilai kepercayaan terhadap gejala, nama penyakit, dan memilih kategori dimana gejala tersebut akan disimpan. Kemudian admin dan pakar diminta untuk menekan tombol simpan setelah memasukkan data gejala yang dibutuhkan, maka sistem akan mengecek apakah nama gejala yang dimasukkan sudah ada di dalam database atau tidak, jika sudah ada di dalam database maka sistem memberi peringatan bahwa nama gejala sudah ada, jika belum ada maka data yang dimasukkan tadi disimpan di dalam database. Tombol batal digunakan untuk kembali ke menu kelola data gejala.

4.9. Antarmuka Halaman Ubah Data Gejala



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk menambahkan data gejala. Pada saat proses ubah data gejala, admin dan pakar diminta untuk memasukkan nama gejala, nilai kepercayaan terhadap gejala, nama penyakit, dan memilih kategori yang baru dimana gejala tersebut akan disimpan. Kemudian admin dan pakar diminta untuk menekan tombol simpan setelah memasukkan data gejala yang dibutuhkan, maka data gejala yang baru tersebut akan disimpan didalam database. Tombol batal digunakan untuk kembali ke menu kelola data gejala.

4.10. Antarmuka Halaman Kelola Data Solusi



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mengelola data solusi. Admin dan pakar dapat melihat data solusi yang tersimpan pada database, mencari data solusi, menambahkan data solusi, mengubah data solusi, dan menghapus data solusi.

4.11. Antarmuka Halaman Tambah Ubah Data Solusi



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk menambahkan atau mengubah data solusi yang tersimpan di dalam database. Pada saat admin dan pakar ingin menambahkan atau mengubah data solusi, admin dan pakar diminta untuk memilih nama penyakit dan memasukkan solusi terhadap penyakit yang telah dipilih. Setelah admin dan pakar selesai memasukkan data yang dibutuhkan, maka admin dan pakar diminta untuk menekan tombol simpan. Pada saat proses tambah data solusi, maka sistem akan mengecek apakah nama penyakit yang dimasukkan sudah ada di dalam database atau tidak, jika sudah ada di dalam database maka sistem memberi peringatan bahwa nama penyakit sudah ada, jika belum ada maka data yang dimasukkan tadi disimpan di dalam database. Pada saat proses ubah data, maka data yang dimasukkan tadi akan disimpan di dalam database.

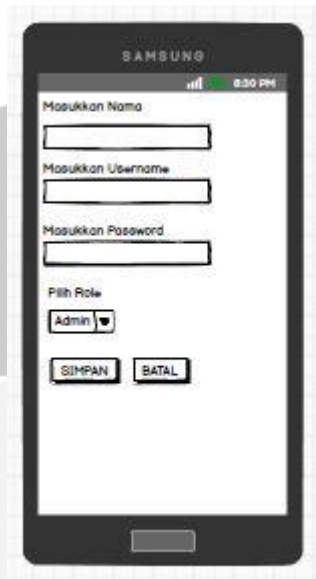
Tombol batal digunakan untuk kembali ke menu kelola data solusi.

4.12. Antarmuka Halaman Kelola Data User



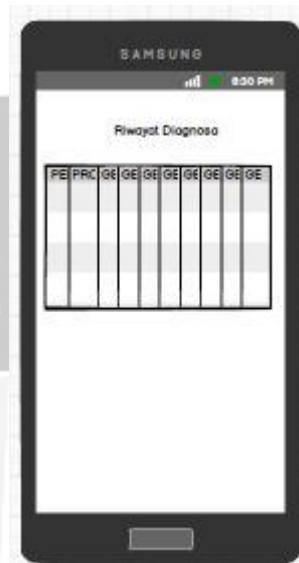
Antarmuka tersebut dapat digunakan oleh admin untuk mengelola data pengguna SiPaPa. Admin dapat melihat data pengguna yang tersimpan pada database, mencari data pengguna, menambahkan data pengguna, mengubah data pengguna, dan menghapus data pengguna.

4.13. Antarmuka Halaman Tambah Ubah Data User



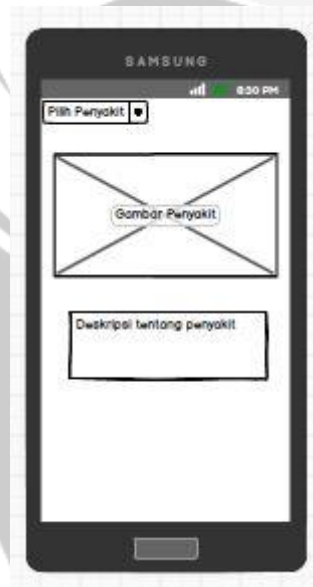
Antarmuka tersebut dapat digunakan oleh admin untuk menambahkan atau mengubah data pengguna SiPaPa yang tersimpan di dalam database. Pada saat admin ingin menambahkan atau mengubah data pengguna, admin diminta untuk memasukkan nama, username, password, dan memilih role untuk user baru yang ingin ditambahkan ke dalam data base. Setelah admin selesai memasukkan data yang dibutuhkan, maka admin diminta untuk menekan tombol simpan. Pada saat proses tambah data pengguna, maka sistem akan mengecek apakah username yang dimasukkan sudah ada di dalam database atau tidak, jika sudah ada di dalam database maka sistem memberi peringatan bahwa username sudah ada, jika belum ada maka data yang dimasukkan tadi disimpan di dalam database. Pada saat proses ubah data, maka data yang dimasukkan tadi akan disimpan di dalam database. Tombol

4.15. Antarmuka Halaman Kelola Riwayat Untuk User



Antarmuka tersebut dapat digunakan oleh *user* untuk melihat riwayat hasil diagnosa yang telah dilakukan sebelumnya.

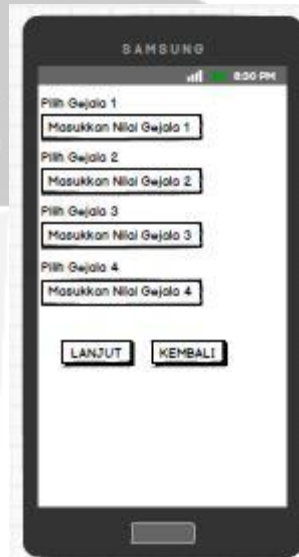
4.16. Antarmuka Halaman Informasi



Antarmuka tersebut dapat digunakan oleh *user* untuk melihat informasi mengenai macam-

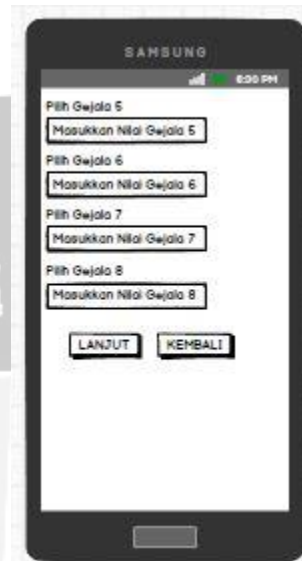
macam penyakit yang terdapat pada organ pankreas manusia.

4.17. Antarmuka Halaman Diagnosa Pertama



Antarmuka tersebut dapat digunakan oleh user untuk melakukan diagnosa dengan memilih gejala dari gejala 1 sampai 4 dan user diminta unruk memasukkan nilai setiap gejala yang telah dipilih. Tombol lanjut digunakan untuk melanjutkan diagnosa berikutnya. Tombol kembali digunakan untuk kembali ke antarmuka menu user.

4.18. Antarmuka Halaman Diagnosa Kedua



Antarmuka tersebut dapat digunakan oleh user untuk melanjutkan diagnosa. User diminta untuk memilih gejala dari gejala 1 sampai 4 dan user diminta untuk memasukkan nilai setiap gejala yang telah dipilih. Tombol lanjut digunakan untuk mengetahui hasil diagnosa. Tombol kembali digunakan untuk kembali ke antarmuka menu diagnosa pertama.

4.19. Antarmuka Halaman Hasil Diagnosa



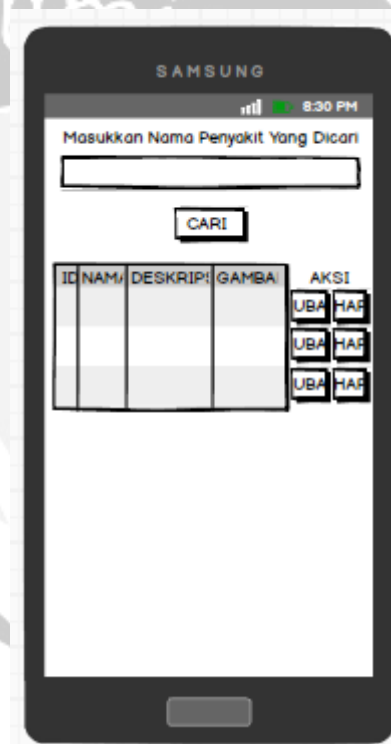
Antarmuka tersebut dapat digunakan oleh user untuk mengetahui hasil diagnosa. Hasil diagnosa meliputi nama penyakit, prosentase user menderita penyakit tersebut, deskripsi penyakit, solusi penyakit, dan gejala yang telah dipilih oleh user.

4.20. Antarmuka Halaman Kategori



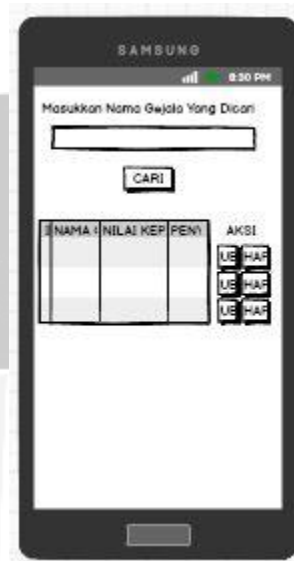
Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk melihat data gejala, mengubah data gejala, mencari data gejala, dan mencari data gejala.

4.21. Antarmuka Halaman Cari Data Penyakit



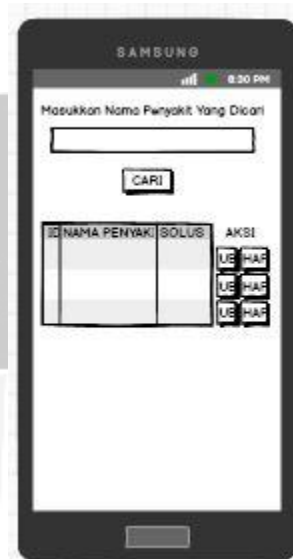
Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mencari data penyakit. Dalam mencari data penyakit, admin dan pakar diminta untuk memasukkan nama penyakit, kemudian menekan tombol cari. Setelah menekan tombol cari maka sistem akan menampilkan data penyakit yang dicari. Dalam antarmuka ini, data penyakit yang dicari dapat diubah atau dihapus.

4.22. Antarmuka Halaman Cari Data Gejala



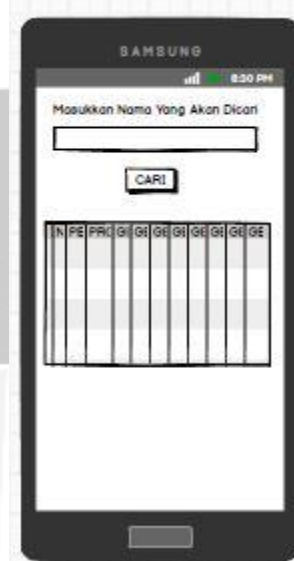
Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mencari data gejala. Dalam mencari data gejala, admin dan pakar diminta untuk memasukkan nama gejala, kemudian menekan tombol cari. Setelah menekan tombol cari maka sistem akan menampilkan data gejala yang dicari. Dalam antarmuka ini, data gejala yang dicari dapat diubah atau dihapus.

4.23. Antarmuka Halaman Cari Data Solusi



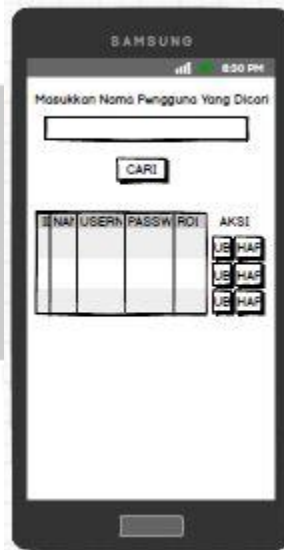
Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mencari data solusi. Dalam mencari data solusi, admin dan pakar diminta untuk memasukkan nama penyakit, kemudian menekan tombol cari. Setelah menekan tombol cari maka sistem akan menampilkan data solusi yang dicari. Dalam antarmuka ini, data solusi yang dicari dapat diubah atau dihapus.

4.24. Antarmuka Halaman Cari Data Riwayat



Antarmuka tersebut dapat digunakan oleh admin dan pakar untuk mencari data riwayat. Dalam mencari data riwayat, admin dan pakar diminta untuk memasukkan nama pengguna SiPaPa, kemudian menekan tombol cari. Setelah menekan tombol cari maka sistem akan menampilkan data riwayat yang dicari. Dalam antarmuka ini, data riwayat yang dicari dapat diubah atau dihapus.

4.25. Antarmuka Halaman Cari Data User



Antarmuka tersebut dapat digunakan oleh admin untuk mencari data pengguna SiPaPa. Dalam mencari data pengguna SiPaPa, admin diminta untuk memasukkan nama pengguna SiPaPa, kemudian menekan tombol cari. Setelah menekan tombol cari maka sistem akan menampilkan data pengguna yang dicari. Dalam antarmuka ini, data pengguna yang dicari dapat diubah atau dihapus.

4.26. Antarmuka Halaman Daftar



Antarmuka tersebut dapat digunakan oleh user yang belum terdaftar dalam SiPaPa. User untuk dapat masuk ke dalam SiPaPa harus melakukan pendaftaran. Pendaftaran dilakukan dengan mengisi nama, username, password.

4.27. Antarmuka Halaman Ubah Password



Antarmuka tersebut dapat digunakan oleh pakar dan user untuk mengubah password. Dalam

antarmuka ini pakar dan user dapat mengubah password lama menjadi baru, dengan mengisikan password baru dan konfirmasi password. Tombol simpan digunakan sistem untuk mengecek apakah password baru sama dengan konfirmasi password, jika sama maka password baru akan disimpan di dalam database, jika tidak maka sistem akan menampilkan pesan password tidak sesuai.



SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SiPaPa

(Sistem Pakar Mendiagnosa Penyakit Pada
Organ Pankreas Manusia Menggunakan Metode
Certainty Factor)


Untuk:

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Lukas Kurniawan/130707190

Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta
2017

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		<i>SKPL-SiPaPa</i>		1/39
	Fakultas Teknologi Industri	<i>Revisi</i>		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh	LK							
Diperiksa oleh	MRT							
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

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	16
2.4	Batasan-batasan	16
2.5	Asumsi dan Ketergantungan	17
3	Kebutuhan khusus	17
3.1	Kebutuhan antarmuka eksternal	17
3.2	Antarmuka pemakai	17
3.3	Antarmuka perangkat keras	18
4	Spesifikasi Rinci Kebutuhan	19
4.1	Spesifikasi Kebutuhan Fungsionalitas	19
5	Entity Relationship Diagram	39

Daftar Gambar

Gambar 1	Arsitektur Perangkat Lunak SiPaPa	11
Gambar 2	Use Case Diagram Untuk Mobile	19
Gambar 3	Gambar Entity Relationship Diagram	39



1. Pendahuluan

1.1. Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SiPaPa (Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor) 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 dan mendefinisikan kebutuhan perancangan perangkat lunak.

1.2. Lingkup Masalah

Perangkat Lunak SiPaPa dikembangkan dengan tujuan untuk:

1. Memudahkan pengguna dalam mendapatkan informasi mengenai penyakit pada organ pankreas manusia.
2. Melakukan pengelolaan seperti penambahan, pengubahan, penghapusan terhadap data penyakit, gejala, solusi, pengguna yang dikelola oleh admin dan pakar.
3. Memudahkan pengguna mengetahui solusi mengenai penyakit pada organ pankreas manusia.

Dan berjalan pada lingkungan dengan platform *Mobile*.

Program Studi Teknik Informatika	SKPL – SiPaPa	6/39
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.
SKPL-SiPaPa-XXX	Kode yang merepresentasikan kebutuhan pada SiPaPa (Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor), dimana XXX merupakan nomor fungsi produk.
SKPL-SiPaPa-XXX-YY	Kode yang merepresentasikan kebutuhan pada SiPaPa (Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor), dimana XXX merupakan nomor fungsi produk dan YY merupakan nomor sub produk.
Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas	Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia adalah sebuah sistem pakar yang

Manusia Menggunakan Metode Certainty Factor (SiPaPa)	digunakan untuk mendiagnosa, menampilkan informasi, dan solusi mengenai penyakit yang terdapat pada organ pankreas manusia
Aplikasi Mobile	Aplikasi yang berjalan pada perangkat mobile.
Hak Akses	Hak untuk masuk dalam mengelola data pada SiPaPa
Login	Proses autentifikasi untuk memastikan bahwa yang menggunakan aplikasi adalah pengguna yang memiliki hak akses.
Data	Kumpulan fakta yang belum memiliki arti. Dalam kasus ini yang berhubungan dengan mendiagnosa penyakit pada SiPaPa.
Basis Data	Tempat penyimpanan data.
Sistem Operasi / OS	Perangkat lunak <i>mobile</i> yang bertugas melakukan kontrol dan manajemen perangkat keras dan juga operasi-operasi dasar sistem.

1.4. Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Lukas Kurniawan, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ILUSI*, Universitas Atma Jaya Yogyakarta.
2. Aditya Budiman, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) BRsErs*, Universitas Atma Jaya Yogyakarta.

1.5. Deskripsi umum (Overview)

Secara umum, dokumen SKPL ini terdiri dari 5 bagian utama. Yang pertama adalah bagian yang berisi penjelasan mengenai dokumen SKPL ini yang mencakup tentang tujuan pembuatan dokumen SKPL, ruang lingkup masalah pengembangan perangkat lunak SiPaPa, definisi dari singkatan dan akronim yang dipakai, referensi penulisan SKPL, dan deskripsi umum tentang dokumen SKPL ini. Bagian kedua berisi penjelasan umum tentang perangkat lunak yang akan dikembangkan (SiPaPa), yang 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. Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SiPaPa yang akan dikembangkan. Bagian keempat berisi penjelasan spesifikasi kebutuhan fungsionalitas mengenai alur proses fungsionalitas sistem. Bagian kelima merupakan *Entity Relationship Diagram* perangkat lunak SiPaPa yang menunjukkan bahwa sistem memiliki 12 entitas

Program Studi Teknik Informatika	SKPL – SiPaPa	9/39
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		

yaitu: gejala1, gejala2, gejala3, gejala4, gejala5, gejala6, gejala7, gejala8, penyakit, solusi, user, riwayat. *Entity Relationship Diagram* menunjukkan hubungan antar tabel yang digunakan oleh aplikasi SiPaPa.

2. Deskripsi Kebutuhan

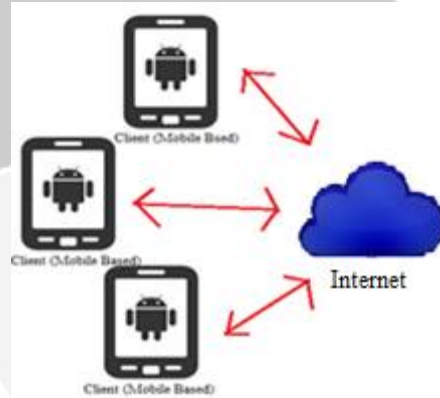
2.1. Perspektif produk

SiPaPa (Sistem Pakar Mendiagnosa Penyakit Pada Organ Pankreas Manusia Menggunakan Metode Certainty Factor) merupakan perangkat lunak yang dibangun untuk membantu pengguna dalam mendiagnosa penyakit pada organ pankreas, informasi mengenai macam-macam penyakit pada organ pankreas, dan solusi terhadap penyakit yang terjadi pada organ pankreas. Sistem ini dibangun dalam aplikasi *mobile* yang dapat dipergunakan oleh pihak SiPaPa sendiri dan pihak pengguna.

Aplikasi *mobile* yang dikembangkan adalah aplikasi yang akan digunakan oleh pengguna yang ingin mengetahui macam-macam penyakit pada organ pankreas dan mendiagnosa penyakit pada organ pankreasnya. Aplikasi ini dibangun untuk menangani pengelolaan data penyakit pada organ pankreas, pengelolaan data solusi, pengelolaan data gejala, melakukan diagnosa, pengelolaan data pengguna, melihat data riwayat, melihat informasi penyakit, dan mengubah password. Untuk alasan keamanan, maka pihak SiPaPa untuk mengelola data penyakit pada organ pankreas yang

menggunakan aplikasi *mobile* ini harus melakukan login.

Perangkat lunak mobile SiPaPa ini dibangun dengan bahasa pemrograman Java dan dapat berjalan pada sistem operasi Android.



Gambar 1. Arsitektur Sistem SiPaPa

2.2. Fungsi Produk

Fungsi produk perangkat lunak SiPaPa adalah sebagai berikut:

a. Fungsi login (**SKPL-SiPaPa-001**).

Merupakan fungsi yang digunakan oleh Admin, Pakar, dan User untuk masuk kedalam SiPaPa.

b. Fungsi mengubah password (**SKPL-SiPaPa-002**).

Merupakan fungsi yang digunakan oleh pihak Pakar dan User dari SiPaPa untuk mengubah password.

c. Fungsi pengelolaan data penyakit pankreas (**SKPL-SiPaPa-003**).

Merupakan fungsi yang digunakan oleh pihak Admin dan Pakar dari SiPaPa.

Fungsi pengelolaan data penyakit pankreas terdiri dari:

1. Fungsi memasukkan data penyakit pankreas **(SKPL-SiPaPa-003-01)** .

Merupakan fungsi yang digunakan untuk menambahkan data penyakit pankreas kedalam SiPaPa.

2. Fungsi menghapus data penyakit pankreas **(SKPL-SiPaPa-003-02)** .

Merupakan fungsi yang digunakan untuk menghapus data penyakit pankreas yang terdapat pada SiPaPa.

3. Fungsi mengubah data penyakit pankreas **(SKPL-SiPaPa-003-03)** .

Merupakan fungsi yang digunakan untuk mengubah data penyakit pankreas yang terdapat pada SiPaPa.

4. Fungsi menampilkan data penyakit pankreas **(SKPL-SiPaPa-003-04)** .

Merupakan fungsi yang digunakan untuk menampilkan data penyakit pankreas yang terdapat pada SiPaPa.

5. Fungsi mencari data penyakit pankreas **(SKPL-SiPaPa-003-05)** .

Merupakan fungsi yang digunakan untuk mencari data penyakit pankreas yang terdapat pada SiPaPa.

d. Fungsi pengelolaan data pengguna **(SKPL-SiPaPa-004)** .

Merupakan fungsi yang digunakan oleh pihak Admin dari SiPaPa.

Fungsi pengelolaan data pengguna terdiri dari:

Program Studi Teknik Informatika	SKPL – SiPaPa	12/39
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. Fungsi memasukkan data pengguna (**SKPL-SiPaPa-004-01**).

Merupakan fungsi yang digunakan untuk menambahkan data pengguna kedalam SiPaPa.

2. Fungsi menghapus data pengguna (**SKPL-SiPaPa-004-02**).

Merupakan fungsi yang digunakan untuk menghapus data pengguna yang terdapat pada SiPaPa.

3. Fungsi mengubah data pengguna (**SKPL-SiPaPa-004-03**).

Merupakan fungsi yang digunakan untuk mengubah data pengguna yang terdapat pada SiPaPa.

4. Fungsi menampilkan data pengguna (**SKPL-SiPaPa-004-04**).

Merupakan fungsi yang digunakan untuk menampilkan data pengguna yang terdapat pada SiPaPa.

5. Fungsi mencari data pengguna (**SKPL-SiPaPa-004-05**).

Merupakan fungsi yang digunakan untuk mencari data pengguna yang terdapat pada SiPaPa.

e. Fungsi pengelolaan data gejala (**SKPL-SiPaPa-005**).

Merupakan fungsi yang digunakan oleh pihak Admin dan Pakar dari SiPaPa.

Fungsi pengelolaan data gejala terdiri dari:

1. Fungsi memasukkan data gejala (**SKPL-SiPaPa-005-01**).

Merupakan fungsi yang digunakan untuk menambahkan data gejala kedalam SiPaPa.

2. Fungsi menghapus data gejala (**SKPL-SiPaPa-005-02**).

Merupakan fungsi yang digunakan untuk menghapus data gejala yang terdapat pada SiPaPa.

3. Fungsi mengubah data gejala (**SKPL-SiPaPa-005-03**).

Merupakan fungsi yang digunakan untuk mengubah data gejala yang terdapat pada SiPaPa.

4. Fungsi menampilkan data gejala (**SKPL-SiPaPa-005-04**).

Merupakan fungsi yang digunakan untuk menampilkan data gejala yang terdapat pada SiPaPa.

5. Fungsi mencari data gejala (**SKPL-SiPaPa-005-05**).

Merupakan fungsi yang digunakan untuk mencari data gejala yang terdapat pada SiPaPa.

f. Fungsi pengelolaan data solusi (**SKPL-SiPaPa-006**).

Merupakan fungsi yang digunakan oleh pihak Admin dan Pakar dari SiPaPa.

Fungsi pengelolaan data solusi terdiri dari:

1. Fungsi memasukkan data solusi (**SKPL-SiPaPa-006-01**).

Merupakan fungsi yang digunakan untuk menambahkan data solusi kedalam SiPaPa.

2. Fungsi menghapus data solusi (**SKPL-SiPaPa-006-02**).

Merupakan fungsi yang digunakan untuk menghapus data solusi yang terdapat pada SiPaPa.

3. Fungsi mengubah data solusi (**SKPL-SiPaPa-006-03**).

Merupakan fungsi yang digunakan untuk mengubah data solusi yang terdapat pada SiPaPa.

4. Fungsi menampilkan data solusi (**SKPL-SiPaPa-006-04**).

Merupakan fungsi yang digunakan untuk menampilkan data solusi yang terdapat pada SiPaPa.

5. Fungsi mencari data solusi (**SKPL-SiPaPa-006-05**).

Merupakan fungsi yang digunakan untuk mencari data solusi yang terdapat pada SiPaPa.

g. Fungsi melihat informasi penyakit (**SKPL-SiPaPa-007**).

Merupakan fungsi yang digunakan oleh pihak User dari SiPaPa untuk melihat informasi mengenai macam-macam penyakit pada organ pankreas manusia.

h. Fungsi melakukan diagnosa (**SKPL-SiPaPa-008**).

Merupakan fungsi yang digunakan oleh pihak User dari SiPaPa untuk memilih gejala yang terdapat pada SiPaPa dan memasukkan nilai berdasarkan gejala yang telah dipilih.

i. Fungsi pengelolaan data riwayat (**SKPL-SiPaPa-009**).

Merupakan fungsi yang digunakan oleh pihak Admin dan Pakar dari SiPaPa.

Fungsi pengelolaan data riwayat terdiri dari:

1. Fungsi menampilkan data riwayat (**SKPL-SiPaPa-009-01**).

Merupakan fungsi yang digunakan untuk menampilkan data riwayat yang terdapat pada SiPaPa.

2. Fungsi mencari data solusi (**SKPL-SiPaPa-009-02**).

Merupakan fungsi yang digunakan untuk mencari data riwayat yang terdapat pada SiPaPa.

j. Fungsi riwayat

Fungsi yang digunakan user untuk melihat riwayat berdasarkan diagnosa sebelumnya yang dilakukan pengguna tersebut.

2.3. Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SiPaPa adalah sebagai berikut:

- a. Memahami pengoperasian sistem operasi Android
- b. Mengerti perangkat lunak yang digunakan

2.4. Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak SiPaPa adalah sebagai berikut:

- a. Aplikasi yang dibangun adalah aplikasi yang akan digunakan untuk melakukan diagnosa dan

mengetahui informasi mengenai penyakit pada organ pankreas manusia

b. Aplikasi dari sistem ini memberikan kesimpulan berupa jenis gejala, solusi, dan penyakit pada organ pankreas manusia.

Sistem pakar ini memberikan kesimpulan berupa jenis penyakit pada organ pankreas manusia berdasarkan data gejala yang dimasukkan oleh pengguna.

2.5. Asumsi dan Ketergantungan

Aplikasi Mobile pada sistem ini dapat dijalankan pada perangkat mobile yang menggunakan sistem operasi Android.

3. Kebutuhan Khusus

3.1. Kebutuhan antarmuka eksternal

Kebutuhan antarmuka eksternal pada perangkat lunak SiPaPa meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, dan antarmuka komunikasi.

3.2. Antarmuka pemakai

Pengguna berinteraksi dengan perangkat lunak SiPaPa dengan antarmuka halaman aplikasi berbasis *mobile*. Masukan dari perangkat lunak ini adalah pilihan menu yang dilakukan dengan memilih pilihan menu yang dilakukan dengan memilih pilihan menu yang ada. Keluaran dari perangkat lunak SiPaPa berupa kesimpulan diagnosa dengan dasar pilihan gejala dari pengguna.

Program Studi Teknik Informatika	SKPL – SiPaPa	17/39
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		

3.3. Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan untuk berinteraksi antara pengguna dengan sistem adalah:

- a. Perangkat *Mobile* dengan sistem operasi Android, digunakan untuk menampilkan informasi kepada pengguna, mengenali *input* yang dilakukan oleh pengguna, untuk mengenali *inputan* yang digunakan oleh pengguna untuk memasukkan data berupa teks, dan angka. *Mobile* dapat digunakan untuk memproses data.
- b. Perangkat Server Basis Data
- c. Dll

3.3.1. Antarmuka perangkat lunak

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

- a. Nama : Android
Sumber : Google
Sebagai sistem operasi yang digunakan.
- b. Nama : Android Studio
Sumber : Google
Sebagai tools pemrograman

3.3.2. Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIPA menggunakan protocol WAP.

3.4. Kebutuhan Fungsionalitas Perangkat Lunak

3.4.1. Use Case Diagram



Gambar 2. Use Case Diagram untuk aplikasi Mobile

4. Spesifikasi Rinci Kebutuhan

4.1. Spesifikasi Kebutuhan Fungsionalitas

4.1.1. Use Case Spesification: Mengelola Data Penyakit Pankreas

Program Studi Teknik Informatika	SKPL – SiPaPa	19/39
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. Brief Description

Use case ini digunakan untuk mengelola data penyakit pankreas. Aktor dapat melakukan penambahan data penyakit pankreas, mengubah data penyakit pankreas, menghapus data penyakit pankreas, mencari data penyakit pankreas, dan melihat data penyakit pankreas.

2. Primary Actor

- a. Admin
- b. Pakar

3. Supporting Actor

-

4. Basic Flow

a. Use case ini dimulai ketika aktor memilih menu untuk pengelolaan data penyakit pankreas.

b. Aktor memilih untuk menambah data penyakit pankreas.

A-1 Aktor memilih untuk mengubah data penyakit pankreas.

A-2 Aktor memilih untuk menghapus data penyakit pankreas.

A-3 Aktor memilih untuk menampilkan data penyakit pankreas

A-4 Aktor memilih untuk mencari data penyakit pankreas.

c. Aktor menginputkan data penyakit pankreas.

d. Aktor meminta sistem untuk menyimpan data penyakit pankreas yang telah dimasukkan.

Program Studi Teknik Informatika	SKPL – SiPaPa	20/39
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		

e. Sistem melakukan pengecekan data penyakit pankreas.

E-1 Data penyakit pankreas yang dimasukkan tidak lengkap

E-2 Nama penyakit pankreas yang dimasukkan sudah ada didalam basis data.

f. Sistem memasukkan data penyakit pankreas kedalam basis data.

g. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk mengubah data penyakit pankreas.

a. Sistem menampilkan data penyakit pankreas.

b. Aktor memilih data penyakit pankreas yang hendak diubah.

c. Aktor mengubah data penyakit pankreas yang sudah dipilih.

d. Aktor meminta sistem untuk menyimpan data penyakit pankreas yang sudah diubah.

e. Sistem melakukan pengecekan terhadap data penyakit pankreas yang dimasukkan.

E-3 Data penyakit pankreas yang diubah aktor tidak lengkap.

f. Sistem menyimpan data penyakit pankreas yang telah diubah kedalam basis data.

g. Berlanjut ke Basic Flow langkah ke-g.

A-2 Aktor memilih untuk menghapus data penyakit pankreas.

a. Sistem menampilkan data penyakit pankreas yang ada.

- b. Aktor memilih data penyakit pankreas yang hendak dihapus.
 - c. Aktor meminta sistem untuk menghapus data penyakit pankreas yang sudah dipilih.
 - d. Sistem menghapus data penyakit pankreas yang diminta oleh aktor.
 - e. Berlanjut ke Basic Flow langkah ke-g.
- A-3 Aktor memilih untuk menampilkan data penyakit pankreas.
- a. Sistem menampilkan seluruh data penyakit pankreas.
 - b. Berlanjut ke Basic Flow langkah ke-g.
- A-4 Aktor memilih untuk mencari data penyakit pankreas.
- a. Sistem meminta aktor untuk memasukkan nama penyakit pankreas yang ingin dicari
 - b. Sistem menampilkan data penyakit pankreas yang dicari.
 - c. Berlanjut ke Basic Flow langkah ke-g.

6. Error Flow

- E-1 Data penyakit pankreas yang dimasukkan tidak lengkap.
- a. Sistem memberi peringatan bahwa data yang diinputkan tidak lengkap.
 - b. Kembali ke Basic Flow langkah ke-d.
- E-2 Nama penyakit pankreas yang dimasukkan sudah ada didalam basis data.
- a. Sistem memberi peringatan bahwa nama penyakit pankreas yang diinputkan sudah terdapat didalam basis data.

b. Kembali ke Basic Flow langkah ke-c.

E-3 Data penyakit pankreas yang telah diubah tidak lengkap.

a. Sistem memberi peringatan bahwa data yang diubah salah.

b. Kembali ke Alternative Flow A-1 langkah ke-b.

7. PreConditions

a. Use case Login telah dilakukan.

8. PostConditions

a. Data penyakit pankreas telah diperbarui.

4.1.2. Use Case Spesification: Mengelola Data Gejala

1. Brief Description

Use case ini digunakan untuk mengelola data gejala penyakit pankreas. Aktor dapat melakukan penambahan data gejala penyakit pankreas, mengubah data gejala penyakit pankreas, menghapus data gejala penyakit pankreas, mencari data gejala penyakit pankreas, dan melihat data gejala penyakit pankreas.

2. Primary Actor

- a. Admin
- b. Pakar

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu untuk pengelolaan data gejala penyakit pankreas.
- b. Aktor memilih untuk menambah data gejala penyakit pankreas.
 - A-1 Aktor memilih untuk mengubah data gejala penyakit pankreas.
 - A-2 Aktor memilih untuk menghapus data gejala penyakit pankreas.
 - A-3 Aktor memilih untuk menampilkan data gejala penyakit pankreas
 - A-4 Aktor memilih untuk mencari data gejala penyakit pankreas.
- c. Aktor menginputkan data gejala penyakit pankreas.
- d. Aktor meminta sistem untuk menyimpan data gejala penyakit pankreas yang telah dimasukkan.
- e. Sistem melakukan pengecekan data gejala penyakit pankreas.
 - E-1 Data gejala penyakit pankreas yang dimasukkan tidak lengkap
 - E-2 Nama gejala penyakit pankreas yang dimasukkan sudah ada didalam basis data.
- f. Sistem memasukkan data gejala penyakit pankreas kedalam basis data.
- g. Use case selesai.

5. Alternative Flow

- A-1 Aktor memilih untuk mengubah data gejala penyakit pankreas.

Program Studi Teknik Informatika	SKPL – SiPaPa	24/39
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		

- a. Sistem menampilkan data gejala penyakit pankreas.
 - b. Aktor memilih data gejala penyakit pankreas yang hendak diubah.
 - c. Aktor mengubah data gejala penyakit pankreas yang sudah dipilih.
 - d. Aktor meminta sistem untuk menyimpan data gejala penyakit pankreas yang sudah diubah.
 - e. Sistem melakukan pengecekan terhadap data gejala penyakit pankreas yang dimasukkan. E-3 Data gejala penyakit pankreas yang diubah aktor tidak lengkap.
 - f. Sistem menyimpan data gejala penyakit pankreas yang telah diubah kedalam basis data.
 - g. Berlanjut ke Basic Flow langkah ke-g.
- A-2 Aktor memilih untuk menghapus data gejala penyakit pankreas.
- a. Sistem menampilkan data gejala penyakit pankreas yang ada.
 - b. Aktor memilih data gejala penyakit pankreas yang hendak dihapus.
 - c. Aktor meminta sistem untuk menghapus data gejala penyakit pankreas yang sudah dipilih.
 - d. Sistem menghapus data gejala penyakit pankreas yang diminta oleh aktor.
 - e. Berlanjut ke Basic Flow langkah ke-g.
- A-3 Aktor memilih untuk menampilkan data gejala penyakit pankreas.

a. Sistem menampilkan seluruh data gejala penyakit pankreas.

b. Berlanjut ke Basic Flow langkah ke-g.

A-4 Aktor memilih untuk mencari data gejala penyakit pankreas.

a. Sistem meminta aktor untuk memasukkan nama gejala penyakit pankreas yang ingin dicari

b. Sistem menampilkan data gejala penyakit pankreas yang dicari.

c. Berlanjut ke Basic Flow langkah ke-g.

6. Error Flow

E-1 Data gejala penyakit pankreas yang dimasukkan tidak lengkap.

a. Sistem memberi peringatan bahwa data yang diinputkan tidak lengkap.

b. Kembali ke Basic Flow langkah ke-d.

E-2 Nama gejala penyakit pankreas yang dimasukkan sudah ada didalam basis data.

a. Sistem memberi peringatan bahwa nama gejala penyakit pankreas yang diinputkan sudah terdapat didalam basis data.

b. Kembali ke Basic Flow langkah ke-c.

E-3 Data gejala penyakit pankreas yang telah diubah tidak lengkap.

a. Sistem memberi peringatan bahwa data gejala yang diubah salah.

b. Kembali ke Alternative Flow A-1 langkah ke-b.

7. PreConditions

- a. Use case Login telah dilakukan.

8. PostConditions

- a. Data gejala penyakit pankreas telah diperbarui.

4.1.3. Use Case Spesification: Mengelola Data Solusi

1. Brief Description

Use case ini digunakan untuk mengelola data solusi penyakit pankreas. Aktor dapat melakukan penambahan data solusi penyakit pankreas, mengubah data solusi penyakit pankreas, menghapus data solusi penyakit pankreas, mencari data solusi penyakit pankreas, dan melihat data solusi penyakit pankreas.

2. Primary Actor

- a. Admin
- b. Pakar

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu untuk pengelolaan data solusi penyakit pankreas.
- b. Aktor memilih untuk menambah data solusi penyakit pankreas.

A-1 Aktor memilih untuk mengubah data solusi penyakit pankreas.

A-2 Aktor memilih untuk menghapus data solusi penyakit pankreas.

Program Studi Teknik Informatika	SKPL – SiPaPa	27/39
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		

A-3 Aktor memilih untuk menampilkan data solusi penyakit pankreas

A-4 Aktor memilih untuk mencari data solusi penyakit pankreas.

c. Aktor menginputkan data solusi penyakit pankreas.

d. Aktor meminta sistem untuk menyimpan data solusi penyakit pankreas yang telah dimasukkan.

e. Sistem melakukan pengecekan data solusi penyakit pankreas.

E-1 Data solusi penyakit pankreas yang dimasukkan tidak lengkap

E-2 Nama solusi penyakit pankreas yang dimasukkan sudah ada didalam basis data.

f. Sistem memasukkan data solusi penyakit pankreas kedalam basis data.

g. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk mengubah data solusi penyakit pankreas.

a. Sistem menampilkan data solusi penyakit pankreas.

b. Aktor memilih data solusi penyakit pankreas yang hendak diubah.

c. Aktor mengubah data solusi penyakit pankreas yang sudah dipilih.

d. Aktor meminta sistem untuk menyimpan data solusi penyakit pankreas yang sudah diubah.

- e. Sistem melakukan pengecekan terhadap data solusi penyakit pankreas yang dimasukkan.
- E-3 Data solusi penyakit pankreas yang diubah aktor tidak lengkap.
- f. Sistem menyimpan data solusi penyakit pankreas yang telah diubah kedalam basis data.
- g. Berlanjut ke Basic Flow langkah ke-g.

A-2 Aktor memilih untuk menghapus data solusi penyakit pankreas.

- a. Sistem menampilkan data solusi penyakit pankreas yang ada.
- b. Aktor memilih data solusi penyakit pankreas yang hendak dihapus.
- c. Aktor meminta sistem untuk menghapus data solusi penyakit pankreas yang sudah dipilih.
- d. Sistem menghapus data solusi penyakit pankreas yang diminta oleh aktor.
- e. Berlanjut ke Basic Flow langkah ke-g.

A-3 Aktor memilih untuk menampilkan data solusi penyakit pankreas.

- a. Sistem menampilkan seluruh data solusi penyakit pankreas.
- b. Berlanjut ke Basic Flow langkah ke-g.

A-4 Aktor memilih untuk mencari data solusi penyakit pankreas.

- a. Sistem meminta aktor untuk memasukkan nama solusi penyakit pankreas yang ingin dicari

b. Sistem menampilkan data solusi penyakit pankreas yang dicari.

c. Berlanjut ke Basic Flow langkah ke-g.

6. Error Flow

E-1 Data solusi penyakit pankreas yang dimasukkan tidak lengkap.

a. Sistem memberi peringatan bahwa data yang diinputkan tidak lengkap.

b. Kembali ke Basic Flow langkah ke-d.

E-2 Nama solusi penyakit pankreas yang dimasukkan sudah ada didalam basis data.

a. Sistem memberi peringatan bahwa nama solusi penyakit pankreas yang diinputkan sudah terdapat didalam basis data.

b. Kembali ke Basic Flow langkah ke-c.

E-3 Data solusi penyakit pankreas yang telah diubah tidak lengkap.

a. Sistem memberi peringatan bahwa data solusi yang diubah salah.

b. Kembali ke Alternative Flow A-1 langkah ke-b.

7. PreConditions

a. Use case Login telah dilakukan.

8. PostConditions

a. Data solusi penyakit pankreas telah diperbarui.

4.1.4. Use Case Spesification: Mengelola Data Pengguna

1. Brief Description

Use case ini digunakan untuk mengelola data pengguna. Aktor dapat melakukan

Program Studi Teknik Informatika	SKPL – SiPaPa	30/39
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		

penambahan data pengguna, mengubah data pengguna, menghapus data pengguna, mencari data pengguna, dan melihat data pengguna.

2. Primary Actor

- a. Admin

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu untuk pengelolaan data pengguna.
 - b. Aktor memilih untuk menambah data pengguna.
 - A-1 Aktor memilih untuk mengubah data pengguna.
 - A-2 Aktor memilih untuk menghapus data pengguna.
 - A-3 Aktor memilih untuk menampilkan data pengguna
 - A-4 Aktor memilih untuk mencari data pengguna.
 - c. Aktor menginputkan data pengguna.
 - d. Aktor meminta sistem untuk menyimpan data pengguna yang telah dimasukkan.
 - e. Sistem melakukan pengecekan data pengguna.
 - E-1 Data pengguna yang dimasukkan tidak lengkap
 - E-2 Nama pengguna yang dimasukkan sudah ada didalam basis data.
 - f. Sistem memasukkan data pengguna kedalam basis data.

Program Studi Teknik Informatika	SKPL – SiPaPa	31/39
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		

g. Use case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk mengubah data pengguna.

- a. Sistem menampilkan data pengguna.
- b. Aktor memilih data pengguna yang hendak diubah.
- c. Aktor mengubah data pengguna yang sudah dipilih.
- d. Aktor meminta sistem untuk menyimpan data pengguna yang sudah diubah.
- e. Sistem melakukan pengecekan terhadap data pengguna yang dimasukkan.

E-3 Data pengguna yang diubah aktor tidak lengkap.

- f. Sistem menyimpan data pengguna yang telah diubah kedalam basis data.
- g. Berlanjut ke Basic Flow langkah ke-g.

A-2 Aktor memilih untuk menghapus data pengguna.

- a. Sistem menampilkan data pengguna yang ada.
- b. Aktor memilih data pengguna yang hendak dihapus.
- c. Aktor meminta sistem untuk menghapus data pengguna yang sudah dipilih.
- d. Sistem menghapus data pengguna yang diminta oleh aktor.
- e. Berlanjut ke Basic Flow langkah ke-g.

A-3 Aktor memilih untuk menampilkan data pengguna.

a. Sistem menampilkan seluruh data pengguna.

b. Berlanjut ke Basic Flow langkah ke-g.

A-4 Aktor memilih untuk mencari data pengguna.

a. Sistem meminta aktor untuk memasukkan nama pengguna penyakit pankrea yang ingin dicari

b. Sistem menampilkan data pengguna yang dicari.

c. Berlanjut ke Basic Flow langkah ke-g.

6. Error Flow

E-1 Data pengguna yang dimasukkan tidak lengkap.

a. Sistem memberi peringatan bahwa data yang diinputkan tidak lengkap.

b. Kembali ke Basic Flow langkah ke-d.

E-2 Nama pengguna yang dimasukkan sudah ada didalam basis data.

a. Sistem memberi peringatan bahwa nama pengguna yang diinputkan sudah terdapat didalam basis data.

b. Kembali ke Basic Flow langkah ke-c.

E-3 Data pengguna yang telah diubah tidak lengkap.

a. Sistem memberi peringatan bahwa data pengguna yang diubah salah.

b. Kembali ke Alternative Flow A-1 langkah ke-b.

7. PreConditions

- a. Use case Login telah dilakukan.

8. PostConditions

- a. Data pengguna telah diperbarui.

4.1.5. Use Case Spesification: Melakukan Diagnosa

1. Brief Description

Use case ini digunakan oleh aktor untuk mendiagnosa penyakit pada organ pankreas yang sedang diderita

2. Primary Actor

- a. User

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu diagnosa.
- b. Sistem menampilkan gejala mengenai penyakit pada organ pankreas manusia.
- c. Aktor memilih gejala yang telah disediakan oleh sistem
- d. Aktor memasukkan nilai berdasarkan gejala yang dipilih
- e. Sistem melakukan proses Certainty Factor berdasarkan gejala yang telah dipilih dan nilai yang telah dimasukkan oleh user.
- f. Sistem mendapatkan hasil nama, deskripsi, solusi mengenai penyakit pada organ pankreas berdasarkan gejala yang telah dipilih oleh user, dan gejala yang telah dipilih oleh user.

g. Sistem menampilkan nama, deskripsi, solusi mengenai penyakit pada organ pankreas berdasarkan gejala yang telah dijawab oleh user, dan menampilkan gejala yang telah dipilih oleh user.

h. Use case selesai.

5. Alternative Flow

-

6. Error Flow

-

7. PreConditions

a. Aktor telah memasuki sistem

8. PostConditions

a. Laporan informasi nama, deskripsi, solusi mengenai penyakit pada organ pankreas berdasarkan gejala yang telah dipilih oleh user, dan gejala yang telah dipilih oleh user.

4.1.6. Use Case Spesification: Melihat Informasi Penyakit.

1. Brief Description

Use case ini digunakan oleh aktor untuk melihat informasi macam-macam penyakit pada organ pankreas manusia

2. Primary Actor

a. User

3. Supporting Actor

-

4. Basic Flow

a. Use case ini dimulai ketika aktor memilih menu informasi.

- b. Sistem menampilkan macam-macam penyakit pada organ pankreas.
- c. Aktor memilih salah satu penyakit pada organ pankreas
- d. Sistem menampilkan deskripsi dan solusi mengenai penyakit pada organ pankreas yang dipilih oleh aktor
- e. Use case selesai.

5. Alternative Flow

-

6. Error Flow

-

7. PreConditions

- a. Aktor telah memasuki sistem

8. PostConditions

- a. Laporan informasi macam-macam penyakit pada organ pankreas

4.1.7. Use Case Spesification: Melihat Data Riwayat.

1. Brief Description

Use case ini digunakan oleh aktor untuk melihat informasi riwayat diagnosa

2. Primary Actor

- a. Admin
- b. Pakar

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu riwayat.

b. Sistem menampilkan data riwayat diagnosa seluruh pengguna sistem .

c. Use case selesai.

5. Alternative Flow

-

6. Error Flow

-

7. PreConditions

a. Aktor telah memasuki sistem

8. PostConditions

a. Laporan informasi data riwayat diagnosa.

4.1.8. Use Case Spesification: Mengubah Password

1. Brief Description

Use case ini digunakan untuk mengubah password.

2. Primary Actor

a. Pakar

b. User

3. Supporting Actor

-

4. Basic Flow

a. Use case ini dimulai ketika aktor memilih menu untuk ubah password.

b. Aktor menginputkan password baru.

c. Aktor meminta sistem untuk menyimpan password baru.

d. Sistem memasukkan password baru kedalam basis data.

e. Use case selesai.

5. Alternative Flow

-

6. Error Flow

-

7. PreConditions

- a. Use case Login telah dilakukan.

8. PostConditions

- a. Password baru telah diperbarui.

4.1.9. Use Case Spesification: Melihat data Riwayat Tertentu.

1. Brief Description

Use case ini digunakan oleh aktor untuk melihat informasi riwayat diagnosa yang dimiliki pengguna

2. Primary Actor

- a. User

3. Supporting Actor

-

4. Basic Flow

- a. Use case ini dimulai ketika aktor memilih menu riwayat diagnosa.
- b. Sistem menampilkan data riwayat diagnosa yang dimiliki pengguna.
- c. Use case selesai.

5. Alternative Flow

-

6. Error Flow

-

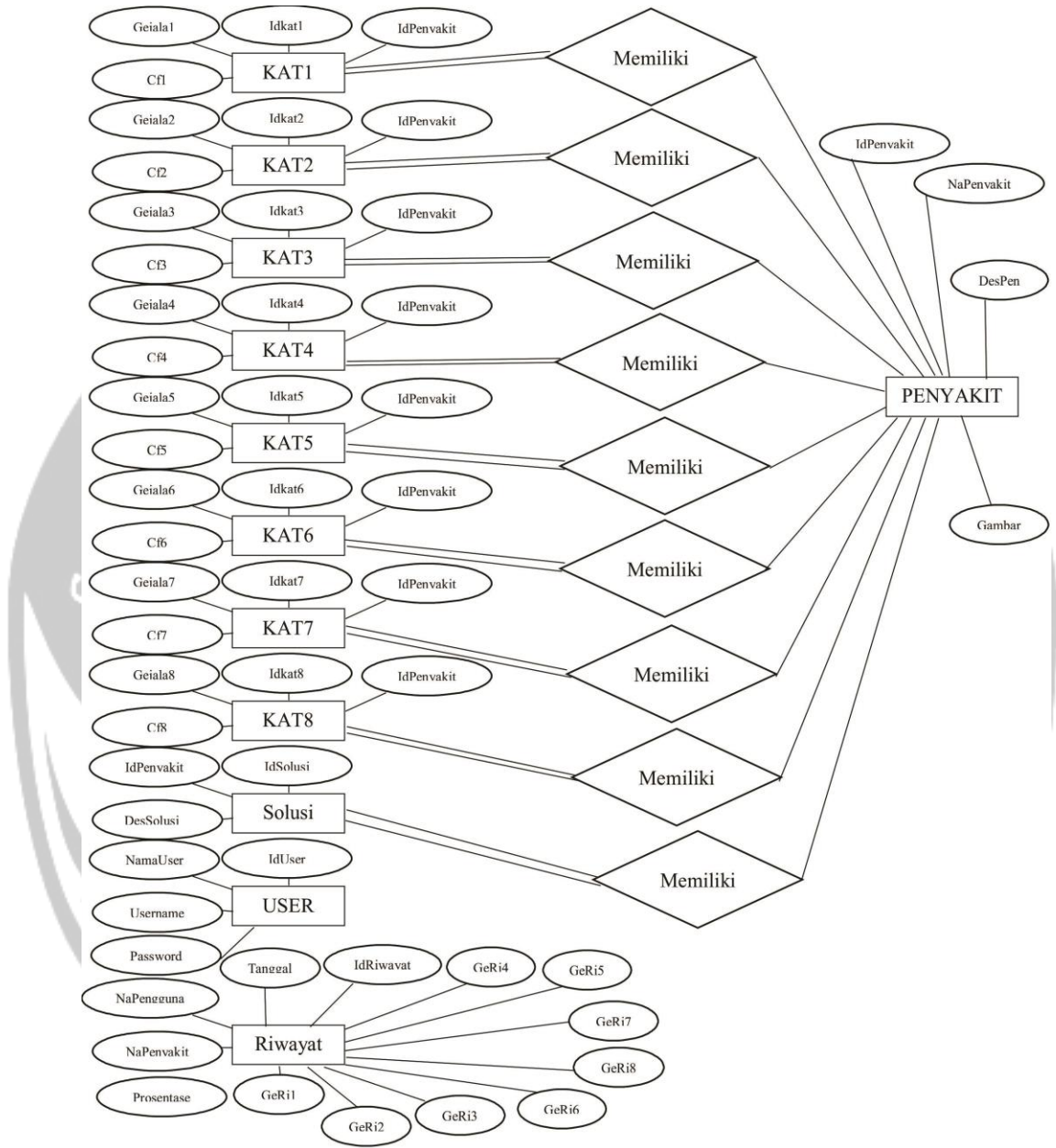
7. PreConditions

- a. Aktor telah memasuki sistem

8. PostConditions

- a. Laporan informasi data riwayat diagnosa.

Entity Relationship Diagram (ERD)



Gambar 3. Gambar Entity Relationship Diagram