

**PENGEMBANGAN LANJUT SISTEM INFORMASI KESEHATAN
“ANDAL” BERDASARKAN MODEL ORGANISASI DAN
MANAJEMEN KESEHATAN PRIMER “ANDAL” DENGAN
PENDEKATAN *INCREMENTAL PROCESS MODEL***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Fathir Izzuddin Qisthi
NIM: 135150400111006



PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2017

DAFTAR ISI

DAFTAR ISI	ii
DAFTAR TABEL	v
DAFTAR GAMBAR	vii
DAFTAR LAMPIRAN	ix
BAB 1 PENDAHULUAN	10
1.1 Latar Belakang	10
1.2 Rumusan Masalah	11
1.3 Tujuan	12
1.3.1 Tujuan Umum	12
1.3.2 Tujuan Khusus	12
1.4 Manfaat	12
1.5 Batasan Masalah	12
1.6 Sistematika Pembahasan	12
BAB 2 LANDASAN KEPUSTAKAAN	14
2.1 Penelitian Terdahulu	14
2.2 Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”	14
2.2.1 Visi dan Misi	15
2.2.2 Tujuan	16
2.2.3 Strategi	16
2.2.4 Produk Pelayanan Kesehatan Primer	17
2.3 Sistem Informasi Kesehatan “ANDAL”	17
2.4 Proses Bisnis	18
2.4.1 Analisis dan Pemodelan Proses Bisnis As-Is	19
2.4.2 Analisis dan Pemodelan Proses Bisnis To-Be	19
2.5 Business Process Modeling Notation (BPMN)	20
2.5.1 Flow Objects	20
2.5.2 Data	22
2.5.3 Connecting Objects	22
2.5.4 Swimlanes	23
2.5.5 Artifacts	23

2.6 Pemodelan Use Case	24
2.6.1 Identifikasi Tipe Pemangku Kepentingan	24
2.6.2 Analisis Masalah.....	25
2.6.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna .	26
2.6.4 Identifikasi Fitur	26
2.6.5 Identifikasi Persyaratan Fungsional dan Persyaratan Non-Fungsional.....	27
2.6.6 Pemodelan Use Case.....	27
2.7 Unified Modeling Language (UML)	28
2.7.1 Use Case Diagram	28
2.7.2 Activity Diagram.....	30
2.7.3 Sequence Diagram	32
2.7.4 Class Diagram.....	35
2.8 Physical Data Model	37
BAB 3 METODOLOGI	38
3.1 Alur Penelitian	38
3.1.1 Studi Pustaka	39
3.1.2 Analisis Kebutuhan.....	39
3.1.3 Pengumpulan Data.....	39
3.1.4 Incremental Process.....	40
3.1.5 Penarikan Kesimpulan.....	41
BAB 4 ANALISIS PERSYARATAN	42
4.1 Pemodelan Proses Bisnis	42
4.1.1 Proses Bisnis As-Is	42
4.1.2 Proses Bisnis To-be	45
4.2 Analisis Persyaratan.....	49
4.2.1 Identifikasi Tipe Pemangku Kepentingan	49
4.2.2 Analisis Masalah.....	50
4.2.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna .	52
4.2.4 Identifikasi Pengguna	58
4.2.5 Identifikasi Fitur	59
4.2.6 Persyaratan Fungsional	61

4.3 Pemodelan Use Case	63
4.3.1 Use Case Diagram	63
4.3.2 Deskripsi Aktor	65
4.3.3 Spesifikasi Use Case	66
4.4 Activity Model	98
4.4.1 Activity Diagram Mengelola Data Pengobatan Holistik	99
4.4.2 Activity Diagram Membaca Panduan Cara Menghindari dan Mengatasi Penyakit	99
4.4.3 Activity Diagram Menghitung Tingkat Risiko Penyakit dan Tingkat Stres Pasien	99
4.4.4 Activity Diagram Mencatat Data Obat Keluar	102
BAB 5 PERANCANGAN DAN IMPELEMANTASI	103
5.1 Perancangan Sequence Diagram	103
5.1.1 Sequence Diagram Mengelola Data Pengobatan Holistik	103
5.1.2 Sequence Diagram Membaca Panduan Cara Menghindari dan Mengatasi Penyakit	107
5.1.3 Sequence Diagram Menghitung Tingkat Risiko Penyakit dan Tingkat Stres Pasien	109
5.2 Perancangan Class Diagram	111
5.2.1 Perancangan Analysis Class Diagram	112
5.2.2 Perancangan Controller Class Diagram	112
5.2.3 Perancangan Model Class Diagram	112
DAFTAR PUSTAKA	117

DAFTAR TABEL

Tabel 2.1 Notasi Events	20
Tabel 2.2 Notasi Activity	21
Tabel 2.3 Notasi Gateways	21
Tabel 2.4 Notasi Data	22
Tabel 2.5 Notasi Connecting Objects	22
Tabel 2.6 Notasi Swimlanes	23
Tabel 2.7 Notasi Artifacts	23
Tabel 2.8 Kerangka Dokumentasi Pernyataan Masalah	25
Tabel 2.9 Contoh Tabel Fitur	26
Tabel 2.10 Format Dokumentasi Spesifikasi Use Case	28
Tabel 2.11 Notasi pada Use Case Diagram	29
Tabel 2.12 Notasi pada Activity Diagram	30
Tabel 2.13 Notasi pada Sequence Diagram	33
Tabel 2.14 Notasi pada Class Diagram	36
Tabel 4.1 Tipe Pemangku Kepentingan	49
Tabel 4.2 Problem Statement 1	50
Tabel 4.3 Problem Statement 2	51
Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna	52
Tabel 4.5 Hasil Identifikasi Pengguna	58
Tabel 4.6 Hasil Identifikasi Fitur	59
Tabel 4.7 Hubungan Kebutuhan Pengguna dengan Fitur	60
Tabel 4.8 Persyaratan Fungsional	61
Tabel 4.9 Hubungan Use Case dengan Pemangku Kepentingan	64
Tabel 4.10 Hubungan Use Case dengan Fitur	64
Tabel 4.11 Deskripsi Aktor	65
Tabel 4.12 Spesifikasi Use Case Login	66
Tabel 4.13 Spesifikasi Use Case Mengelola Data Dasar Kesehatan Keluarga	67
Tabel 4.14 Spesifikasi Use Case Mengelola Data Pendaftaran Pengobatan Pasien	76
Tabel 4.15 Spesifikasi Use Case Mengelola Data Anamnesis	79

Tabel 4.16 Spesifikasi Use Case Mengelola Data Pengobatan Holistik	83
Tabel 4.17 Spesifikasi Use Case Membaca Panduan Menghindari dan Mengatasi Penyakit.....	89
Tabel 4.18 Spesifikasi Use Case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien	91
Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat	92
Tabel 4.20 Spesifikasi Use Case Mengelola Data Wilayah	96

DAFTAR GAMBAR

Gambar 2.1 Contoh Use Case Diagram.....	29
Gambar 2.2 Contoh Activity Diagram	30
Gambar 2.3 Contoh Sequence Diagram.....	33
Gambar 2.4 Contoh Physical Data Model	37
Gambar 3.1 Alur Penelitian	38
Gambar 4.1 Proses Bisnis As-Is Pelayanan Kesehatan	43
Gambar 4.2 Proses Bisnis As-Is Pencatatan Obat Keluar	44
Gambar 4.3 Proses Bisnis To-Be Pencatatan Obat Keluar.....	46
Gambar 4.4 Proses Bisnis To-Be Pelayanan Kesehatan	47
Gambar 4.5 Proses Bisnis To-Be Pencatatan Data Dasar Kesehatan Keluarga.....	48
Gambar 4.6 Keterangan Kodifikasi Kebutuhan Pengguna	52
Gambar 4.7 Kodifikasi Fitur	59
Gambar 4.8 Kodifikasi Persyaratan Fungsional.....	61
Gambar 4.9 Use-Case Diagram Sistem Informasi Kesehatan “ANDAL”	63
Gambar 4.10 Activity Diagram Mengelola Data Pengobatan Holistik.....	100
Gambar 4.11 Activity Diagram Membaca Panduan Cara Menghindari dan Mengatasi Penyakit	101
Gambar 4.12 Activity Diagram Menghitung Tingkat Risiko Penyakit dan Tingkat Stres Pasien	101
Gambar 4.13 Activity Diagram Mencatat Data Obat Keluar	102
Gambar 5.1 Sequence Diagram Mengelola Data Pengobatan Holistik	104
Gambar 5.2 Sequence Diagram Memuat Data Modul	105
Gambar 5.3 Sequence Diagram Memuat Detail Data Pengobatan Holistik	105
Gambar 5.4 Sequence Diagram Memproses Penyimpanan Data Pengobatan Holistik.....	106
Gambar 5.5 Sequence Diagram Membaca Panduan Cara Menghindari dan Mengatasi Penyakit	107
Gambar 5.6 Sequence Diagram Memilih Modul Lain	108
Gambar 5.7 Sequence Diagram Menghitung Tingkat Risiko Penyakit dan Tingkat Stres Pasien	109
Gambar 5.8 Sequence Diagram Menampilkan Detail Data Dasar Kesehatan Keluarga.....	110

Gambar 5.9 Sequence Diagram Melakukan Perhitungan Ulang Tingkat Risiko Penyakit dan Tingkat Stres Pasien	111
Gambar 5.10 Controller Class Diagram.....	112
Gambar 5.11 Model Class Diagram	113
Gambar 5.12 Analysis Class Diagram	114
Gambar 5.13 Physical Data Model.....	115

DAFTAR LAMPIRAN

No table of figures entries found.

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” merupakan model yang berisi panduan strategis dan operasional pelayanan kesehatan primer bagi Fasilitas Kesehatan Tingkat Pertama (FKTP) yang terdiri dari puskesmas, praktik dokter, praktik dokter gigi, klinik pratama termasuk fasilitas kesehatan tingkat pertama milik TNI/POLRI, rumah sakit kelas D pratama untuk melaksanakan pengelolaan dalam pengendalian mutu dan biaya pelayanan. Model manajemen tersebut dirancang agar FKTP dapat memberikan pelayanan kesehatan primer kepada sasaran pelayanan (perorangan, keluarga, komunitas peserta BPJS) dengan tujuan memenuhi 5 tujuan dasar pelayanan kesehatan primer secara optimal di era Jaminan Kesehatan Nasional (JKN).

Saat ini Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” diterapkan di Poliklinik Pabrik Gula Kebon Agung, Kecamatan Pakisaji, Kabupaten Malang. Salah satu penerapan model tersebut ialah dengan melaksanakan 10 usaha pokok pada Poliklinik Pabrik Gula Kebon Agung. Berdasarkan hasil wawancara dengan dr. Jack Roebijoso, dalam penerapan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” dibutuhkan dukungan teknologi yang selaras dalam proses pelaksanaannya. Dibutuhkan sistem informasi kesehatan yang mampu membantu mengelola data dasar kesehatan keluarga, mengelola data registrasi dan status pasien, menampilkan tingkat resiko kesehatan dan tingkat stres, mengelola hasil diagnosis pasien, mengelola data tindakan medis yang diberikan kepada pasien, mengelola data rujukan pasien, mengelola data pengecekan darah pasien, mengelola data kegiatan edukasi, advokasi, dan pengelolaan lingkungan tempat tinggal atau tempat kerja pasien. Selain itu, sistem informasi kesehatan diharapkan mampu menampilkan persentase penyakit yang tersebar mulai dari tingkat RT, RW, kelurahan, kecamatan, hingga kota/kabupaten, sehingga dapat membantu menentukan persediaan obat, tindakan medis, edukasi, advokasi, dan pengelolaan lingkungan tempat tinggal atau tempat kerja di tiap wilayah.

Sistem Informasi Kesehatan “ANDAL” saat ini telah dikembangkan dan mampu memberikan fitur pengelolaan data dasar kesehatan keluarga untuk memenuhi kebutuhan pengelolaan data dasar kesehatan keluarga dari masyarakat, fitur registrasi pasien untuk memenuhi kebutuhan pengelolaan data registrasi, fitur menampilkan tingkat resiko kesehatan dan tingkat stres, fitur pengelolaan hasil pemeriksaan paramedis yang meliputi pemeriksaan keluhan pasien, tekanan darah, denyut nadi, suhu badan untuk memenuhi kebutuhan pengelolaan status pasien. Sistem Informasi Kesehatan “ANDAL” yang dikembangkan saat ini merupakan purwarupa dari sistem informasi kesehatan yang dibutuhkan untuk mendukung penerapan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” di Poliklinik Pabrik Gula Kebon Agung, Kabupaten Malang.

Masalah utama dalam sistem informasi kesehatan adalah kegagalan untuk memahami ekspektasi dan persyaratan pengguna (Aggarwal, 2002). Aggarwal menambahkan, kegagalan dalam memahami kebutuhan pengguna menyebabkan kegagalan implementasi dan menyebabkan fungsi sistem informasi kesehatan tidak efisien, sehingga menurut Aggarwal, analis harus mampu menggali persyaratan pengguna dari pernyataan kebutuhan pengguna. Salah satu metode yang dapat digunakan untuk menganalisis masalah dan menentukan persyaratan sistem informasi adalah *Object-oriented Analysis* (OOA) (Aggarwal, 2002). Aggarwal menjelaskan bahwa OOA lebih cenderung menghasilkan spesifikasi persyaratan yang lengkap karena OOA memodelkan sistem seperti objek aslinya.

Menurut Aggarwal, aktivitas yang dapat dilakukan untuk mencapai tujuan OOA diantara lain adalah memodelkan fungsi sistem dan mengidentifikasi objek-objek sistem, memodelkan objek-objek dan mengidentifikasi hubungan antar objek, serta memodelkan *behavior* objek. Setiap aktivitas yang dilakukan akan menghasilkan representasi persyaratan sistem yang berbeda-beda. Pemodelan object-oriented mengkombinasikan tiga gambaran sistem dalam bentuk pemodelan objek, *dynamic*, dan fungsional (Krol & Reich, 1999). Representasi sistem tersebut dapat digambarkan melalui Unified Modeling Language (Aggarwal, 2002). Aggarwal menambahkan bahwa UML menjamin efektivitas komunikasi antara pengguna dengan analis, sehingga dapat memudahkan analis mendokumentasikan persyaratan pengguna.

Setelah analisis dan pemodelan dilakukan, proses implementasi sistem dengan konsep *object-oriented* dapat dimulai dengan menggunakan hasil pemodelan objek yang berisi atribut dan *behavior* objek. Menurut Jacobson, objek-objek yang telah diidentifikasi dan dimodelkan dapat diimplementasi menjadi *class* yang memiliki variabel sebagai representasi atribut objek, serta memiliki *operation* sebagai representasi *behavior* objek. Selain itu Jacobson menambahkan bahwa implementasi harus memperhatikan konsep-konsep enkapsulasi objek, inheritas antar kelas, *class* dan *instances*, serta *polymorphism*.

Berdasarkan hasil wawancara mengenai kebutuhan dukungan sistem informasi kesehatan terhadap penerapan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” di Poliklinik Pabrik Gula Kebon Agung, penelitian ini bertujuan untuk menghasilkan purwarupa pengembangan lanjut Sistem Informasi Kesehatan “ANDAL” dengan menyajikan hasil dokumentasi analisis persyaratan, perancangan, pengembangan lanjut, dan pengujian purwarupa Sistem Informasi Kesehatan “ANDAL” dengan menggunakan pendekatan *object-oriented*.

1.2 Rumusan Masalah

1. Dapatkan pengembangan lanjut Sistem Informasi Kesehatan Primer “ANDAL” dilakukan dengan menggunakan pendekatan berorientasi objek?

1.3 Tujuan

1.3.1 Tujuan Umum

Melakukan pengembangan lanjut Sistem Informasi Kesehatan Primer “ANDAL” dengan menggunakan pendekatan berorientasi objek.

1.3.2 Tujuan Khusus

1. Mengidentifikasi persyaratan sistem dan merancang Sistem Informasi Kesehatan Primer “ANDAL” berdasarkan hasil perancangan sistem terdahulu dengan menggunakan pendekatan berorientasi objek.
2. Membangun Sistem Informasi Kesehatan Primer “ANDAL” berdasarkan hasil identifikasi persyaratan sistem dan perancangan sistem menggunakan pendekatan berorientasi objek.
3. Mengetahui hasil pengujian Sistem Informasi Kesehatan Primer “ANDAL”.

1.4 Manfaat

Manfaat dari penelitian ini adalah terwujudnya purwarupa pengembangan lanjut sistem informasi kesehatan yang dapat menunjang penerapan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” yang sesuai dengan kebutuhan bisnis pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”. Hasil penelitian dapat digunakan sebagai usulan pengembangan sistem informasi kesehatan di masa yang akan datang, selain itu hasil penelitian dapat dijadikan sumber atau acuan pengembangan sistem informasi kesehatan yang lebih baik.

1.5 Batasan Masalah

Pengembangan lanjut Sistem Informasi Kesehatan “ANDAL” dilakukan berdasarkan proses bisnis yang ada pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” yang diterapkan pada Poliklinik Pabrik Gula Kebon Agung, Kabupaten Malang, dengan langkah awal proses bisnis yang dimulai dari proses pengumpulan data dasar kesehatan keluarga, pengisian data registrasi dan status pasien, pelaksanaan diagnosa holistik kepada pasien. Hasil pengembangan lanjut Sistem Informasi Kesehatan “ANDAL” tidak untuk memenuhi kebutuhan dukungan sistem informasi terhadap proses pengelolaan data kegiatan edukasi, advokasi, dan pengelolaan lingkungan tempat tinggal atau tempat kerja pasien.

1.6 Sistematika Pembahasan

Sistematika penulisan penelitian ini antara lain sebagai berikut:

BAB I Pendahuluan

Bab ini berisi latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, serta sistematika pembahasan skripsi.

Bab II Landasan Kepustakaan

Bab ini berisi tinjauan pustaka dari beberapa penelitian sebelumnya, serta berisi pembahasan tentang teori, model, metode, atau sistem berdasarkan literatur ilmiah yang dapat menunjang penulisan penelitian.

BAB III Metodologi

Dalam bab ini menjelaskan langkah-langkah yang dilakukan untuk menyelesaikan penelitian.

BAB IV Analisis Kebutuhan

Bab ini berisi pembahasan kebutuhan sistem yang sesuai dengan hasil identifikasi proses bisnis pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”.

BAB V Perancangan dan Implementasi

Bab ini menjelaskan hasil perancangan dengan menyertakan diagram-diagram yang dibutuhkan sebagai syarat perancangan. Selain itu, bab ini membahas detail implementasi dari fungsi-fungsi yang menjadi fokus utama.

BAB VI Pengujian

Bab ini berisi proses dan hasil pengujian SIKES “ANDAL” dengan menggunakan metode pengujian *functional test* untuk mengetahui apakah fungsi-fungsi sudah berjalan dengan baik *dan user acceptance test* untuk mengetahui tingkat kesesuaian hasil pengembangan dengan kebutuhan bisnis dan sistem.

BAB VII Penutup

Bab ini berisi tentang kesimpulan dan saran berdasarkan hasil pengujian yang dilakukan terhadap hasil pengembangan.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Penelitian Terdahulu

Penelitian ini dilakukan berdasarkan penelitian terdahulu yang membahas penggunaan pendekatan *Object-Oriented Analysis* (OOA), *Object-Oriented Design* (OOD), dan *Object-Oriented Programming* (OOP) dalam analisis, perancangan, dan implementasi perangkat lunak. Penelitian terdahulu juga menjadi landasan peneliti untuk menggunakan *Use-Case Modeling* sebagai bagian dari aktivitas OOA dan OOD untuk menganalisis kebutuhan dan persyaratan, serta kemudian memodelkannya dengan memanfaatkan kombinasi dari beberapa *Unified Modeling Language* (UML). Hasil pemodelan perancangan tersebut kemudian akan menjadi *input* untuk melakukan implementasi sistem informasi kesehatan yang akan dibangun dalam penelitian ini.

Vinod Aggarwal dalam penelitian yang berjudul "*The Application of the Unified Modeling Language in Object-Oriented Analysis of Healthcare Information Systems*" menjelaskan bahwa *Object-Oriented Analysis* merupakan teknik yang sesuai untuk mengembangkan sistem informasi kesehatan. Aktivitas OOA yang dilakukan dalam penelitian tersebut menghasilkan representasi persyaratan sistem informasi dari berbagai sudut pandang, kemudian representasi persyaratan tersebut dapat disajikan dengan Unified Modeling Language (UML). Penelitian tersebut juga menjelaskan bahwa UML memiliki manfaat dalam menentukan, memvisualisasi, membangun, mendokumentasikan, dan mengkomunikasikan hasil dari aktivitas OOA.

Marina Krol dan David L. Reich dalam penelitian yang berjudul "*Object-Oriented Analysis and Design of a Health Care Management Information System*" menjelaskan bahwa Object-Oriented Analysis and Design (OOAD) memiliki keuntungan dalam pengembangan sistem komputer yang kompleks. OOAD juga terbukti diterima oleh komunitas informatika medis untuk mengembangkan sistem kesehatan, salah satunya informasi rumah sakit. Penelitian ini memiliki fokus untuk memodelkan objek, memodelkan urutan operasi masing-masing objek didalam sistem, dan memodelkan fungsi sistem dengan pendekatan OOAD.

Odeh, et al., dalam penelitiannya berjudul "*A Use-Case Driven Approach in Requirements Engineerin: The MAMMOGRID Project*" menunjukkan penerapan teknik *Use-Case Modeling* untuk mengidentifikasi dan menentukan persyaratan pengguna dari proyek MAMMOGRID dalam pendekatan *incremental* dan *iterative*. Penelitian tersebut menunjukkan bahwa teknik Use-Case Modeling memperpendek jarak diantara setiap fase dalam proses Requirement Engineering, sehingga dapat mengurangi siklus proses *Requirement Engineering*.

2.2 Model Organisasi dan Manajemen Kesehatan Primer "ANDAL"

Model Organisasi dan Manajemen Kesehatan Primer "ANDAL" diusulkan berdasarkan pelayanan kesehatan primer di Indonesia masih berorientasi kuratif sehingga dapat menyebabkan pengeluaran biaya yang tinggi dalam pengobatan.

Persentase pengeluaran nasional sektor kesehatan tahun 2005 adalah sebesar 0,81% dan meningkat menjadi 1,09% pada tahun 2007 (Departemen Kesehatan RI, 2009). Begitu juga dengan anggaran kesehatan, jumlah APBN kesehatan tahun 2004 sebesar Rp 5,54 triliun, kemudian meningkat menjadi Rp 18,75 Triliun pada tahun 2007 (Departemen Kesehatan RI, 2009). Pengeluaran pemerintah untuk kesehatan terus meningkat, namun kontribusi pengeluaran pemerintah untuk kesehatan kecil, yaitu 38% dari total pengeluaran kesehatan (Departemen Kesehatan RI, 2009). Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” pernah diterapkan pada tahun 1998. Selama model manajemen tersebut diterapkan, profil rasio orang berobat dari karyawan menurun hingga 46% dan profil angka kejadian kasus rawat inap di RS menurun hingga 33% (Roebijoso, 2010).

Roebijoso (2010) menjelaskan bahwa Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” dapat memberikan panduan untuk menyusun organisasi pelayanan kesehatan primer dokter keluarga yang fungsional, sehingga dapat menjalankan upaya kesehatan primer bagi perorangan (UKP) yang berintegrasi optimal dengan upaya kesehatan bagi masyarakat (UKM). Selain itu, model manajemen tersebut dapat menjadi panduan untuk menyusun kompetensi, materi dan/atau modul, metode, dan evaluasi pendidikan dan/atau tenaga kesehatan klinik dokter keluarga. Model manajemen tersebut akan menyerap tenaga kerja pelayanan kesehatan primer sehingga secara fungsional, tugas pelayanan kesehatan primer dapat dilaksanakan secara optimal.

Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” merupakan model dengan manajemen yang integratif, strategis, efektif, efisien, berkualitas, merata, dan terjangkau yang disusun berdasarkan motivasi untuk memberikan kepuasan kepada pasien dan untuk memberikan layanan promotif, preventif, dan kuratif dengan mengusahakan kegiatan pelayanan promotif dan preventif lebih banyak dibandingkan pemberian pelayanan kuratif. Motivasi tersebut merupakan pendorong stakeholder dalam memberikan pelayanan kesehatan primer sehingga dinilai dapat menurunkan jumlah rawat inap dan rawat jalan pasien karena layanan pencegahan penyakit lebih banyak diberikan daripada layanan kuratif, sehingga penyakit dapat terobati dan dapat dicegah, yang menyebabkan angka pasien rawat inap dan rawat jalan semakin berkurang. Kepuasan pasien dapat mendorong pasien untuk terdaftar pada puskesmas, klinik, atau pada dokter praktek perorangan sehingga dapat meningkatkan kapitasi dan pendapatan puskesmas, klinik, atau pada dokter praktek perorangan (Nugraha, 2017).

2.2.1 Visi dan Misi

Visi Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” adalah “Menjadi sentuhan pertama untuk dapat hidup sehat”.

Misi Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” adalah “Memberdayakan hidup sehat dan memberikan pelayanan kesehatan dasar pada sasaran”.

2.2.2 Tujuan

Menurut Roebijoso(2010) tujuan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” adalah sebagai berikut:

1. Meningkatkan pemberdayaan hidup sehat dan derajat kesehatan (rasio usaha pokok pencegahan dengan pengobatan).
2. Menurunkan angka kesakitan, kematian, kecacatan, dan kecelakaan.
3. Menurunkan angka rujukan medis dan rawat inap di Rumah Sakit.
4. Menurunkan total pembiayaan kesehatan dan mengoptimalkan biaya kesehatan primer.
5. Meningkatkan kualitas, pemerataan, keterjangkauan pelayanan kesehatan primer serta dapat dievaluasi tiap tribulan, semester, tahun, sampai 5 tahun untuk menilai keberhasilan strategi.

2.2.3 Strategi

Roebijoso (2010) menjelaskan penyusunan alternatif strategi pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” dilakukan dengan memperhatikan hasil kajian model strategi pelayanan dokter keluarga dengan bukti penurunan jumlah kunjungan pasien berobat (*contact rate*) yang dapat digunakan sebagai indikator angka kesakitan, angkar rujukan medis dan rawat inap, serta total biaya kesehatan yang menurun 54% dalam satu tahun. Alternatif strategi pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” adalah sebagai berikut:

1. Menyusun atau modifikasi organisasi UKP yang fungsional dan mampu menjalankan usaha-usaha pokok pelayanan keseshatan dokter keluarga yang dapat diadopsi dari sebuah model manajemen strategis pelayanan kesehatan dokter keluarga (Roebijoso, 2010), yaitu berupa sepuluh usaha-usaha pokok pelayanan kesehatan dokter keluarga, yang terdiri dari 60% upaya promotif preventif (pemberdayaan kesehatan) dan 40% upaya kuratif dasar.
2. Menyusun *standart input*, proses, cakupan *output*, evaluasi *outcome*, dan *impact* dalam manajemen pelayanan kesehatan primer yang integratif, strategis, efektif, efisien, berkualitas, merata, terjangkau yang mengutamakan pencegahan penyakit. Standarisasi *input* meliputi fasilitas personalia terlatih, perangkat lunak, perangkat kerasm teknologi informasi kesehatan, pembiayaan, indikator keberhasilan, materi modul dan sub modul pemberdayaan, dan pelayanan kesehatan medis dasar yang ditujukan bagi perorangan, keluarga, komunitas sasaran yang sesuai dengan kondisi lokal. Standar proses meliputi upaya menentukan prioritas sasaran pelayanan dan *Standart Operating Procedure* (SOP) dari intervensi pelayanan kesehatan primer (medis, non medis) dalam sepuluh usaha-usaha pokok pelayanan kesehatan dokter keluarga dengan modul/sub modul yang baku sehingga kualitas dan konsistensi pelayanan kesehatan primer dapat selalu dijamin karena setiap anggota tim kesehatan akan menjalankan upaya yang dibakukan

di suatu wilayah kerja. Standar evaluasi keberhasilan meliputi alat ukur, waktu, cara mengumpulkan, menyusun dan menampilkan data indikator keberhasilan, melakukan analisa data pelayanan kesehatan, dan rencana intervensi pelayanan kesehatan pada sasaran yang dibakukan agar dapat mencapai tujuan.

2.2.4 Produk Pelayanan Kesehatan Primer

Beberapa usaha pokok pada Upaya Pelayanan Kesehatan bagi Perorangan (UKP) ditambahkan dan terbukti dapat mengurangi angka sakit dan dapat menurunkan biaya pengobatan (Roebijoso, 2010). Usaha pokok pada UKP yang terdiri dari:

1. Diagnosa holistik/komprehensif di klinik.
2. Edukasi dan advokasi kesehatan perorangan di klinik (menggunakan modul).
3. Edukasi dan advokasi kesehatan keluarga di klinik.
4. Tindakan medis (*emergency, diagnostic, kuratif, rehabilitasi dasar*) di klinik.
5. Diagnosa holistik/komprehensif di rumah.
6. Edukasi dan advokasi kesehatan keluarga di rumah.
7. Tindakan medis (*emergency, diagnostic, kuratif, rehabilitasi dasar*) di rumah.
8. Mediasi dan advokasi kesehatan di rumah sakit rujukan.
9. Kunjungan tempat kerja (Identifikasi Resiko K3).
10. Edukasi dan advokasi kesehatan kelompok (komunitas) sesuai dengan masalah kesehatan yang ada.

2.3 Sistem Informasi Kesehatan “ANDAL”

Sistem Informasi Kesehatan “ANDAL” dikembangkan untuk memenuhi kebutuhan dukungan teknologi informasi bagi Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”. Berdasarkan hasil wawancara dengan dr. Jack Roebijoso, Sistem Informasi Kesehatan “ANDAL” pada dasarnya digunakan untuk mengelola data dasar kesehatan keluarga, pengolahan data tersebut digunakan untuk membantu proses pengambilan keputusan tindakan medis bersifat promotif, preventif, dan kuratif, serta digunakan untuk mengevaluasi keberhasilan.

Data dasar kesehatan keluarga berisi identitas keluarga, riwayat penyakit atau kecelakaan yang pernah dialami anggota keluarga, dan data rekam medis pasien, data tersebut akan diperbarui setiap 3 bulan sekali. Data dasar kesehatan keluarga direkam melalui Sistem Informasi Kesehatan “ANDAL” oleh petugas administrasi, petugas rekam medis klinik, tenaga paramedis, atau tenaga medis. Pengolahan data dasar kesehatan keluarga akan menghasilkan tingkat resiko kesehatan dan tingkat stres pada masing-masing anggota keluarga. Sistem Informasi Kesehatan “ANDAL” juga digunakan untuk mengelola data registrasi dan status pasien ketika

pasien ingin mendapatkan pelayanan kesehatan di klinik. Data status pasien berisi hasil pemeriksaan paramedis yang meliputi pemeriksaan keluhan pasien, tekanan darah, denyut nadi, suhu badan, dan lain-lain. Berdasarkan data-data tersebut, tenaga medis dapat mendiagnosis pasien dan merekam hasil diagnosis melalui Sistem Informasi Kesehatan “ANDAL”.

Tenaga medis dapat memberikan tindakan medis berdasarkan modul penyakit yang dapat diakses melalui Sistem Informasi Kesehatan “ANDAL” sesuai dengan hasil diagnosis yang telah direkam oleh tenaga medis. Selain itu, pemberian tindakan medis berdasarkan modul faktor pemicu dan faktor risiko penyakit dapat diberikan kepada pasien oleh tenaga medis setelah data faktor pemicu dan faktor risiko penyakit direkam melalui Sistem Informasi Kesehatan “ANDAL”. Tenaga medis juga dapat mengelola kebutuhan rujukan (rujukan ke rumah sakit atau rujukan tindakan UGD) pasien, mengelola kebutuhan cek darah pasien, serta mengelola kebutuhan resep obat pasien.

Berdasarkan hasil pengolahan data dasar kesehatan keluarga, tenaga medis dapat menentukan kebutuhan kunjungan tempat tinggal dan tempat kerja pasien. Sistem Informasi Kesehatan “ANDAL” dapat membantu tenaga medis dan petugas kesehatan masyarakat untuk mengelola dokumen permintaan pengelolaan lingkungan tempat tinggal atau tempat kerja. Seluruh data hasil kegiatan edukasi kesehatan, advokasi kesehatan, pengelolaan lingkungan tempat tinggal atau tempat kerja akan direkam melalui Sistem Informasi Kesehatan “ANDAL”, seluruh data yang telah direkam melalui Sistem Informasi Kesehatan “ANDAL” digunakan untuk membantu pengambilan keputusan atau kebijakan dalam pelayanan kesehatan primer dan evaluasi keberhasilan dari keputusan atau kebijakan pelayanan kesehatan primer yang diambil.

2.4 Proses Bisnis

Proses bisnis adalah serangkaian aktivitas yang membutuhkan sebuah (atau lebih) *input* dan menghasilkan *output* yang memiliki nilai bagi *customer* (Monk & Wagner, 2013). Monk dan Wagner menambahkan, *customer* sebuah proses bisnis terdiri dari *customer* eksternal dan *customer* internal. *Customer* eksternal merupakan seseorang atau sekelompok orang yang membeli produk, sedangkan *customer* internal merupakan seseorang atau sekelompok orang dalam berbagai macam departemen dalam perusahaan.

Becker, et al. menjelaskan bahwa dalam serangkaian aktivitas dalam perusahaan terdapat serangkaian aktivitas inti (proses inti) dan serangkaian aktivitas pendukung (proses pendukung). Proses inti merupakan proses yang aktivitas didalamnya secara langsung berhubungan dengan produk dari sebuah perusahaan yang akan berkontribusi memberikan nilai (keuntungan) terhadap perusahaan. Sebagai contoh, proses pada bagian pengadaan bahan baku, operasional, pemasaran dan penjualan, logistik dan pelayanan pelanggan yang akan berkontribusi memberikan nilai ekonomis (*profit*) kepada perusahaan.

Proses pendukung berisi serangkaian aktivitas yang tidak memberikan nilai secara langsung kepada pelanggan (eksternal), namun sangat penting dalam pelaksanaan proses inti di dalam perusahaan. Proses bisnis dapat digunakan untuk memahami sebuah organisasi atau perusahaan. Beberapa pendekatan yang dapat dilakukan untuk mengetahui sebuah perusahaan adalah dengan melakukan analisis dan pemodelan terhadap proses bisnis *as-is* dan proses bisnis *to-be*.

2.4.1 Analisis dan Pemodelan Proses Bisnis As-Is

Analisis dan pemodelan proses bisnis *as-is* adalah kegiatan untuk mengumpulkan dan memodelkan proses bisnis yang aktual pada sebuah organisasi atau perusahaan yang bertujuan untuk menyajikan informasi mengenai situasi aktual perusahaan, untuk mengetahui kekurangan dari proses bisnis, dan dapat memunculkan kemungkinan peningkatan dalam proses bisnis yang ada pada sebuah organisasi atau perusahaan (Becker, et al., 2013).

Proses bisnis *as-is* dapat diidentifikasi dengan melakukan wawancara dengan pegawai yang berpartisipasi dalam eksekusi proses bisnis. Selain itu, proses bisnis *as-is* juga dapat diidentifikasi dengan melakukan kegiatan observasi terhadap pegawai yang sedang mengerjakan tugas-tugasnya dalam organisasi atau perusahaan (Przybytek, 2011).

Przybytek menambahkan, hasil identifikasi proses bisnis *as-is* digunakan sebagai skema diagram proses bisnis *as-is*. Diagram proses bisnis *as-is* harus memuat proses fundamental organisasi atau perusahaan. Setiap aktivitas dalam proses bisnis akan dimodelkan, jika terdapat aktivitas yang kompleks dalam sebuah proses bisnis, maka dapat digambarkan sebagai sub-proses dari sebuah proses yang lebih umum. Kemudian proses bisnis *as-is* yang telah dimodelkan harus disetujui oleh pemangku kepentingan.

2.4.2 Analisis dan Pemodelan Proses Bisnis To-Be

Analisis dan pemodelan proses bisnis *to-be* merupakan kegiatan untuk mengidentifikasi dan memodelkan proses bisnis yang telah mengalami peningkatan (*improvement*) berdasarkan analisis terhadap proses bisnis *as-is* (Becker, et al., 2013). Migrasi untuk menerapkan proses bisnis *to-be* dengan menggantikan proses bisnis *as-is* dapat dilakukan secara bertahap, sehingga perubahan proses bisnis tidak akan terasa signifikan secara langsung. Becker, et al. menjelaskan bahwa penerapan proses bisnis *to-be* dapat dilakukan dalam jangka menengah atau jangka panjang.

Analisis dan pemodelan proses bisnis *to-be* dilakukan dengan melibatkan pemangku kepentingan untuk dapat memperoleh kemungkinan peningkatan proses bisnis yang dapat dilakukan pada organisasi atau perusahaan (Przybytek, 2011). Przybytek menjelaskan, peningkatan proses bisnis mungkin dilakukan dengan menerapkan otomatisasi pada beberapa proses bisnis. Namun, Przybytek juga menekankan, menerapkan otomatisasi tidak menjamin peningkatan yang signifikan. Sehingga sangat penting untuk melibatkan pemangku kepentingan dalam kegiatan analisis dan pemodelan proses bisnis *to-be*.

2.5 Business Process Modeling Notation (BPMN)

Object Management Group (OMG) menjelaskan bahwa BPMN merupakan notasi-notasi standar yang digunakan untuk memberikan solusi pemodelan proses bisnis sehingga proses pertukaran informasi antar pengguna bisnis dapat dilakukan dengan baik. BPMN memiliki tujuan untuk menyediakan notasi yang mudah dipahami oleh seluruh pengguna bisnis, mulai dari analis bisnis untuk mempersiapkan konsep proses bisnis, hingga pengguna yang memiliki kepentingan untuk mengelola dan mengawasi bagaimana proses bisnis dijalankan.

BPMN dirancang untuk memberikan mekanisme pemodelan proses bisnis secara sederhana dan mudah dipahami, namun di saat yang bersamaan, BPMN juga mampu menangani visualisasi kompleksitas proses bisnis. Oleh karena itu, BPMN menyediakan sekumpulan notasi yang dibagi menjadi 5 kategori di antaranya adalah *flow objects*, *data*, *connecting objects*, *swimlanes*, *artifacts*.




2.5.1 Flow Objects

Flow object, kategori notasi ini menjelaskan *behavior* yang dimiliki oleh proses bisnis. Kategori ini memiliki 3 elemen yang terdiri dari:

1. Events

Events merupakan notasi yang memberikan pengaruh terhadap alur aktivitas di dalam model BPMN. *Events* pada umumnya adalah notasi yang merepresentasikan *trigger* dan *results* pada sebuah proses bisnis. *Event* terdiri dari 3 tipe, yaitu *start event*, *intermediate event*, dan *end*. Penjelasan notasi *event* terdapat pada Tabel 2.1.

Tabel 2.1 Notasi Events

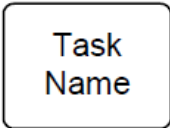
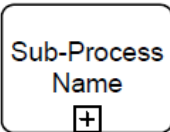
Notasi	Nama	Deskripsi
	<i>Start Event</i>	Notasi yang merepresentasikan sebuah proses dimulai.
	<i>Intermediate Event</i>	Notasi ini muncul di antara <i>start event</i> dan <i>end</i> , serta akan memengaruhi alur proses bisnis. Notasi ini tidak dapat digunakan untuk memulai dan mengakhiri sebuah proses secara langsung. Notasi ini memiliki beberapa macam varian untuk merepresentasikan event yang berbeda-beda. Setiap varian diindikasikan dengan penggunaan simbol-simbol yang berbeda di tengah notasi ini.
	<i>End</i>	Notasi yang merepresentasikan sebuah proses telah selesai.

Sumber: Object Management Group (2011)

2. Activity

Activity merupakan notasi yang digunakan untuk merepresentasikan sebuah pekerjaan yang di dalam sebuah proses. *Activity* memiliki 2 tipe notasi yaitu *sub-process* dan *task*. Penjelasan notasi terdapat pada Tabel 2.2.

Tabel 2.2 Notasi Activity






Notasi	Nama	Deskripsi
	<i>Task</i>	Notasi yang merepresentasikan aktivitas yang bersifat <i>atomic</i> , yang berarti aktivitas tersebut tidak dipecah ke tingkat yang lebih detail.
	<i>Sub-process</i>	Notasi yang digunakan untuk merepresentasikan sekumpulan aktivitas. <i>Sub-process</i> memiliki detail yang berisi sekumpulan aktivitas di dalamnya.

Sumber: Object Management Group (2011)

3. Gateways

Gateways digunakan untuk menjelaskan alur proses bisnis sesuai dengan kondisi yang sedang berjalan. Notasi *gateways* dijelaskan pada Tabel 2.3.

Tabel 2.3 Notasi Gateways






Notasi	Nama	Deskripsi
	<i>Exclusive</i>	Digunakan untuk membuat alur alternatif pada proses bisnis. Dari beberapa alur alternatif yang dibuat, hanya 1 alur alternatif yang dapat dijalankan sesuai dengan kondisi yang terpenuhi.
	<i>Parallel</i>	Digunakan untuk membuat alur paralel dan menyinkronkan alur paralel yang telah dibuat.
	<i>Event-based</i>	Digunakan untuk membuat alur alternatif pada proses bisnis. Alur yang berjalan dipicu oleh sebuah <i>events</i> tertentu yang terjadi.
	<i>Complex</i>	Digunakan untuk memvisualisasikan alur kompleks dalam proses bisnis.
	<i>Inclusive</i>	Digunakan untuk membuat alur alternatif sekaligus dapat berjalan secara paralel pada proses bisnis. Kondisi pada setiap alur diperiksa, kemudian alur yang memenuhi kondisi dijalankan.

Sumber: Object Management Group (2011)

2.5.2 Data

Data digunakan untuk memvisualisasikan bagaimana dokumen, data, dan objek lain digunakan dan diperbarui pada sebuah alur proses (IBM Knowledge Center, 2014). kategori ini direpresentasikan oleh 4 notasi yang terdiri dari *data objects*, *data inputs*, *data outputs*, *data stores*. Masing-masing notasi ditejalskan pada Tabel 2.4.

Tabel 2.4 Notasi Data



Notasi	Nama	Deskripsi
	<i>Data Object</i>	Notasi yang digunakan untuk memberikan informasi tambahan mengenai data atau dokumen atau objek lain yang berkaitan dengan aktivitas atau proses tertentu.
	<i>Data Object (Collection)</i>	Memvisualisasikan <i>data object</i> yang berjumlah lebih dari 1.
	<i>Data Input</i>	Notasi yang digunakan untuk memvisualisasikan data atau dokumen atau objek lain yang menjadi <i>input</i> sebuah atau serangkaian aktivitas.
	<i>Data Output</i>	Notasi yang digunakan untuk memvisualisasikan data atau dokumen atau objek lain yang menjadi <i>output</i> sebuah atau serangkaian aktivitas.
	<i>Data Store</i>	Notasi yang digunakan untuk memberikan informasi mengenai sebuah lokasi penyimpanan dan pembaruan data.

Sumber: Object Management Group (2011)



2.5.3 Connecting Objects

Connecting objects berisi notasi yang digunakan untuk menghubungkan notasi *flow object* dengan notasi-notasi BPMN lainnya. Beberapa notasi yang akan digunakan dalam penelitian ini dijelaskan pada Tabel 2.5.

Tabel 2.5 Notasi Connecting Objects

Notasi	Nama	Deskripsi
	<i>Sequence Flow</i>	Notasi yang digunakan untuk menunjukkan urutan aktivitas yang akan dijalankan di dalam proses bisnis.
	<i>Message Flow</i>	Notasi yang digunakan untuk menunjukkan aliran yang dikirimkan dan diterima sebuah aktivitas yang dimiliki oleh partisipan yang berbeda.

Tabel 2.5 Notasi Connecting Objects



Notasi	Nama	Deskripsi
	<i>Associations</i>	Notasi yang digunakan untuk menghubungkan notasi-notasi <i>artifacts</i> yang berisi informasi dengan notasi-notasi BPMN lainnya.
	<i>Data Associations</i>	Notasi yang digunakan untuk menghubungkan aktivitas BPMN dengan notasi data.

Sumber: Object Management Group (2011)

2.5.4 *Swimlanes*

Swimlanes adalah jenis notasi yang digunakan untuk mengelompokkan aktivitas berdasarkan unit bisnis yang menjalankan aktivitas-aktivitas tertentu. *Swimlanes* pada umumnya digunakan untuk memberikan informasi yang jelas untuk aktivitas *business-to-business* (Object Management Group, 2011). Notasi-notasi *swimlanes* dijelaskan pada Tabel 2.6.


Tabel 2.6 Notasi *Swimlanes*

Notasi	Nama	Deskripsi
	<i>Pool</i>	Notasi yang digunakan untuk merepresentasikan partisipan sebuah proses bisnis. <i>Pool</i> dapat memiliki detail di dalamnya, sehingga seluruh aktivitas yang dimiliki oleh sebuah partisipan dapat dilihat pada model proses bisnis. Namun <i>pool</i> juga dapat bersifat <i>black-box</i> .
	<i>Lane</i>	<i>Lane</i> merupakan sub-partisi dari sebuah <i>pool</i> , notasi ini merepresentasikan unit-unit yang dimiliki oleh sebuah partisipan proses bisnis.

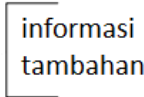
Sumber: Object Management Group (2011)

2.5.5 *Artifacts*

Tabel 2.7 Notasi *Artifacts*

Notasi	Nama	Deskripsi
	<i>Group</i>	Notasi yang digunakan untuk memberikan informasi pengelompokan notasi-notasi kedalam sebuah kategori. Notasi ini tidak memengaruhi alur proses bisnis dan pada umumnya digunakan

Tabel 2.7 Notasi *Artifacts*

Notasi	Nama	Deskripsi
		untuk kepentingan dokumentasi dan analisis. Nama notasi group diletakkan di dalam notasi ini, umumnya diletakkan di sisi bawah notasi.
	<i>Text Annotation</i>	Notasi ini digunakan untuk memberikan informasi dalam bentuk teks untuk memudahkan pembaca BPMN memahami proses bisnis.

Sumber: Object Management Group (2011)

Notasi-notasi *artifacts* digunakan untuk memberikan informasi tambahan pada proses bisnis. Penggunaan *artifacts* tidak memengaruhi alur proses bisnis. Beberapa notasi *artifacts* dijelaskan pada Tabel 2.7.

2.6 Pemodelan *Use Case*

Model *use case* terbukti memiliki nilai berharga sebagai bagian dari aktivitas penentuan persyaratan di dalam *software process*, khususnya dalam meningkatkan komunikasi antara tim pengembang perangkat lunak dengan pemangku kepentingan dan membuat aktivitas penentuan persyaratan lebih mudah dan tepat (Bittner & Spence, 2002). Bittner dan Spence menambahkan, *use case* menggambarkan bagaimana pengguna menggunakan sistem dan apa yang dilakukan sistem untuk penggunaanya.

Pemodelan *use case* dilakukan untuk mengetahui informasi mengenai pemangku kepentingan yang akan terlibat dan terkena dampak dari sebuah sistem, mengetahui siapa saja calon pengguna sistem, mengetahui kebutuhan pemangku kepentingan dan pengguna sistem berdasarkan masalah yang mereka hadapi, serta untuk mengetahui kemungkinan solusi yang dapat ditawarkan kepada pemangku kepentingan dan pengguna untuk memenuhi kebutuhannya.

2.6.1 Identifikasi Tipe Pemangku Kepentingan

Pemangku kepentingan adalah seseorang yang secara materi terkena dampak dari hasil sebuah sistem atau proyek yang memproduksi sebuah sistem (Bittner & Spence, 2002). Identifikasi tipe pemangku kepentingan merupakan aktivitas analisis persyaratan yang digunakan untuk mengetahui siapa saja pemangku kepentingan dari sebuah sistem atau proyek, kemudian pemangku kepentingan tersebut akan dikelompokkan berdasarkan karakteristik dan hubungan pemangku kepentingan dengan sistem atau proyek yang sedang dikerjakan. Bittner dan Spence menjelaskan bahwa terdapat beberapa kategori pemangku kepentingan, yang terdiri dari:

1. Pengguna (*Users*), merupakan pemangku kepentingan yang berinteraksi secara langsung dengan sistem. Pengguna merupakan individu-individu yang akan memerankan peran sebagai aktor yang ada pada model *use case*.
2. Sponsor (*Sponsors*), merupakan pemangku kepentingan yang menanam investasi kedalam pengembangan sistem atau proyek. Pemangku kepentingan ini pada umumnya adalah pengguna tidak langsung dari sistem,
3. Pengembang (*Developers*), merupakan pemangku kepentingan yang terlibat dalam kegiatan produksi dan mendukung pengembangan sistem atau sebuah proyek.
4. Pihak yang berwenang (*Authorities*), merupakan pemangku kepentingan yang memiliki keahlian khusus dalam memberikan dukungan dalam pengembangan sistem dalam bidang regulasi internal atau eksternal sebuah perusahaan, teknologi, aturan perundangan, dan standar organisasi.
5. Customers (*Pelanggan*), merupakan pemangku kepentingan yang akan membeli sistem yang telah selesai dikembangkan.

Namun Bittner dan Spence menekankan bahwa kategori atau tipe pemangku kepentingan tidak terbatas pada kategori-kategori yang sudah dijelaskan sebelumnya. Tipe pemangku kepentingan bisa saja dikembangkan bergantung pada jenis organisasi atau perusahaan, macam-macam calon pengguna sistem, dan sebagainya.

2.6.2 Analisis Masalah

Analisis masalah adalah sebuah proses untuk memahami masalah yang aktual dan kebutuhan pengguna, serta proses untuk mengusulkan solusi untuk memenuhi kebutuhan pengguna (Leffingwell & Widrig, 2003). Leffingwell dan Widrig menjelaskan tujuan analisis masalah adalah untuk mendapatkan pemahaman yang lebih baik mengenai masalah yang akan diselesaikan sebelum proses pengembangan solusi (sistem) dimulai.

Cara terbaik untuk mendokumentasikan analisis masalah adalah dengan membangun pernyataan masalah (*problem statement*) (Bittner & Spence, 2002). Pernyataan masalah diperoleh dengan melakukan wawancara dengan beberapa pemangku kepentingan, kemudian pernyataan masalah dapat didokumentasikan dalam tabulasi sesuai contoh pada Tabel 2.8.

Tabel 2.8 Kerangka Dokumentasi Pernyataan Masalah

<i>The problem of</i>	Berisi deskripsi masalah.
<i>Affects</i>	Berisi pemangku kepentingan yang terpengaruh oleh masalah.
<i>The impact of which is</i>	Berisi deskripsi dampak masalah terhadap pemangku kepentingan dan kegiatan bisnis.

Tabel 2.8 Kerangka Dokumentasi Pernyataan Masalah

<i>A successful solution would</i>	Berisi beberapa solusi dan manfaat utama yang ditawarkan untuk menyelesaikan masalah.
------------------------------------	---

Sumber: Bittner & Spence (2003)

2.6.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Kebutuhan pemangku kepentingan dan pengguna atau secara singkat disebut kebutuhan pengguna adalah sesuatu yang dapat mewakili masalah bisnis, personal, atau operasional yang harus ditekankan dengan tujuan untuk membenarkan alasan untuk mempertimbangkan, membeli, atau menggunakan sistem baru (Leffingwell & Widrig, 2003). Memperoleh informasi mengenai kebutuhan pengguna dapat membantu dalam memahami bagaimana dan sejauh mana beberapa aspek yang berbeda dari sebuah masalah memengaruhi jenis pemangku kepentingan yang berbeda (Bittner & Spence, 2002).

Bittner dan Spence menambahkan bahwa identifikasi kebutuhan pengguna dapat menyediakan pemahaman yang lebih dalam mengenai pernyataan-pernyataan masalah yang disampaikan oleh pemangku kepentingan. Leffingwell dan Widrig menjelaskan, identifikasi kebutuhan pengguna, identifikasi fitur, dan persyaratan perangkat lunak dapat dilakukan dengan beberapa teknik yang terdiri dari:

1. Wawancara
2. *Requirement workshop*
3. *Brainstorming and Idea Reduction*
4. *Storyboarding*

2.6.4 Identifikasi Fitur

Fitur adalah layanan yang disediakan oleh sistem untuk memenuhi satu atau lebih kebutuhan pemangku kepentingan (Leffingwell & Widrig, 2003). Leffingwell dan Widrig menambahkan, fitur merupakan cara berguna dan mudah untuk mendeskripsikan fungsional sistem tanpa harus terhambat dengan penjelasan fungsional sistem yang terlalu detail. Fitur juga menyediakan ringkasan manfaat yang ditawarkan oleh produk (sistem) yang akan dibangun (Bittner & Spence, 2002).

Tabel 2.9 Contoh Tabel Fitur

Kode Fitur	Deskripsi
Berisi kode identitas fitur	Berisi penjelasan tentang yang dapat dilakukan oleh sistem

Sumber: Diadaptasi dari Bittner & Spence (2002)

Bittner dan Spence menjelaskan bahwa penamaan fitur harus memiliki tingkat detail yang umum, sehingga seluruh pemangku kepentingan dapat memahami

kemampuan yang akan diberikan sistem, hal ini ditekankan karena fitur juga digunakan untuk meringkas kemampuan dan kualitas dari sebuah perangkat lunak yang akan dibangun. Fitur yang telah ditentukan dapat didokumentasikan dalam tabulasi sesuai dengan contoh yang terdapat pada Tabel 2.9.

2.6.5 Identifikasi Persyaratan Fungsional dan Persyaratan Non-Fungsional

Bittner dan Spence menyebutkan, persyaratan fungsional dan persyaratan non-fungsional merupakan kategori yang umum digunakan untuk mengelompokkan persyaratan perangkat lunak. Persyaratan fungsional dan non-fungsional dapat diekspresikan dalam banyak cara, namun cara yang paling umum digunakan adalah dengan mendeskripsikan persyaratan-persyaratan tersebut dalam kalimat deklaratif.

Persyaratan fungsional adalah layanan yang sebaiknya dapat disediakan oleh sistem, persyaratan fungsional juga merupakan pernyataan yang menyatakan bagaimana sistem bereaksi terhadap *input* tertentu dan bagaimana sistem berjalan ketika berada pada situasi tertentu (Sommerville, 2011). Deskripsi dari persyaratan fungsional sebaiknya berisi penjelasan mengenai apa tindakan yang dapat dilakukan oleh sebuah sistem. Selain itu didalamnya terdapat informasi perilaku sistem ketika menerima *input* dan *output*.

Persyaratan non-fungsional merupakan aspek kualitas atau batasan-batasan yang harus dipenuhi oleh sistem (Bittner & Spence, 2002). Leffingwell dan Widrig menjelaskan bahwa persyaratan non-fungsional muncul karena sangat penting bagi analis untuk menentukan aspek-aspek sistem yang berhubungan dengan *usability*, *reliability*, *performance*, dan *supportability* dari sebuah sistem.

2.6.6 Pemodelan Use Case

Model *use case* digunakan untuk mempresentasikan sistem dalam hal bagaimana penggunaan sistem tersebut (Bittner & Spence, 2002). Bittner dan Spence menambahkan, dalam menentukan *use case*, analis menggunakan persyaratan perangkat lunak sebagai panduan dan seringkali *use case* berisi persyaratan fungsional dari sebuah sistem. Model *use case* berisi dua komponen utama yaitu aktor dan *use Case*.

Aktor adalah representasi individu (manusia) atau objek lain yang berinteraksi dengan sistem. Aktor merupakan objek diluar sistem. Dalam model *use case*, aktor memiliki nama dan deskripsi singkat, selain itu, aktor dihubungkan dengan *use case* sesuai dengan interaksi aktor dengan *use case* tertentu.

Use case adalah representasi sebuah nilai yang diberikan oleh sistem kepada aktor. *Use case* tidak sama dengan fitur atau fungsi dari sebuah sistem. *Use case* memiliki nama dan penjelasan singkat. *Use case* juga memiliki spesifikasi yang lebih detail, yang berisi alur ketika aktor dan sistem berinteraksi.

Model *use case* tidak menyediakan deskripsi yang lengkap mengenai persyaratan sebuah sistem, oleh karena itu, model *use case* harus didukung

dengan pemodelan-pemodelan persyaratan dan dokumen-dokumen persyaratan lainnya. Setiap *use case* di dalam model *use case* memiliki spesifikasi *use case* yang berisi penjelasan bagaimana aktor dan sistem berkolaborasi untuk memenuhi tujuan yang direpresentasikan oleh *use case* tersebut. Format dokumentasi spesifikasi *use case* terdapat pada Tabel 2.10.

Tabel 2.10 Format Dokumentasi Spesifikasi *Use Case*

Brief Description	Berisi penjelasan singkat sebuah <i>use case</i> .
Actor	Berisi aktor yang berinteraksi dengan <i>use case</i> .
Pre-condition	Berisi satu atau lebih kondisi atau keadaan yang harus dipenuhi oleh sistem sebelum <i>use case</i> dapat dijalankan.
Post-condition	Berisi satu atau lebih kondisi atau keadaan yang dipenuhi oleh sistem ketika <i>use case</i> berakhir.
Extension	Berisi informasi penggunaan <i>use case</i> lain yang bersifat opsional, yang disertakan pada sebuah <i>use case</i> .
Basic Flow	Berisi alur normal yang dilalui ketika <i>use case</i> berjalan.
Alternative Flow	Berisi alur opsional yang akan dilalui ketika <i>use case</i> tidak berjalan sesuai dengan alur normal.
Subflow	Berisi beberapa kelompok alur yang merupakan penyederhanaan alur <i>use case</i> yang terlalu detail.

Sumber: Diadaptasi dari Bittner & Spence (2002)

2.7 Unified Modeling Language (UML)

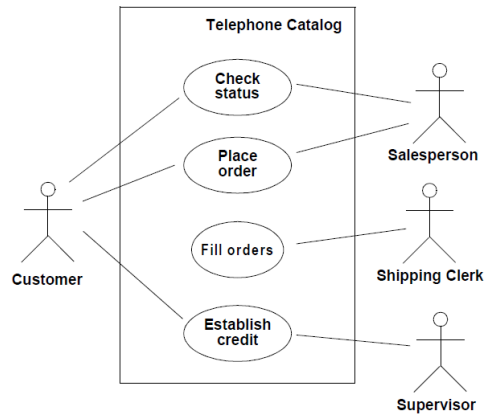
UML adalah bahasa untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan artefak sistem perangkat lunak, pemodelan bisnis, dan sistem non-perangkat lunak lainnya (Object Management Group, 2005). UML berisi beberapa pilihan diagram yang dapat digunakan untuk memvisualisasikan sistem ke dalam beberapa model. Diagram-diagram pada UML terdiri dari *use case diagram*, *class diagram*, *behavior diagram*, *implementation diagram*.

Beberapa diagram tersebut menyediakan beragam sudut pandang bagi analis atau pengembang. Beberapa kombinasi diagram dapat digunakan untuk mengintegrasikan berbagai macam sudut pandang yang berguna untuk pengembangan sistem. Peneliti memanfaatkan beberapa diagram dalam penelitian ini terdiri dari *use case diagram*, *class diagram*, *activity diagram* (varian dari *behavior diagram*), dan *sequence diagram* (varian dari *interaction diagram*, yang juga merupakan varian dari *behavior diagram*).

2.7.1 Use Case Diagram

Use case diagram menampilkan hubungan di antara aktor dengan *use case* (Object Management Group, 2005). Hubungan yang tergambar di dalam *use case*

diagram adalah hubungan asosiasi antara aktor dengan satu atau lebih *use case*, hubungan generalisasi antar aktor, dan hubungan generalisasi, *extends*, dan *includes* antar *use case*. Sekumpulan *use case* secara opsional dapat diletakkan di dalam sebuah notasi berbentuk persegi yang merepresentasikan *boundary* sebuah sistem. Contoh *use case diagram* diperlihatkan dalam Gambar 2.1.

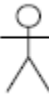
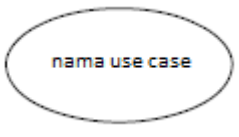
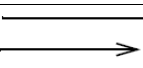
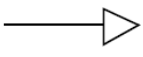


Gambar 2.1 Contoh Use Case Diagram

Sumber: Object Management Group (2005)

Use case diagram memiliki beberapa komponen notasi, di antaranya notasi aktor, notasi *use case*, notasi hubungan *use case*, dan notasi hubungan aktor. Notasi-notasi *use case diagram* dijelaskan pada Tabel 2.11.

Tabel 2.11 Notasi pada Use Case Diagram

Notasi	Nama	Deskripsi
	Aktor	Aktor merupakan peran yang diperankan oleh pengguna atau sistem lain yang berinteraksi dengan sistem. Satu aktor bisa saja berinteraksi dengan satu atau lebih <i>use case</i> .
	<i>Use case</i>	<i>Use case</i> adalah spesifikasi dari sekumpulan aksi yang dilakukan oleh sistem yang akan memberikan nilai untuk satu atau lebih aktor atau pemangku kepentingan sebuah sistem.
	Asosiasi	Notasi hubungan antara aktor dengan <i>use case</i> yang menggambarkan aktor berinteraksi dengan sistem dengan melakukan <i>use case</i> tertentu.
	Generalisasi	Notasi hubungan antar aktor yang menghubungkan aktor yang lebih spesifik dengan aktor yang umum.

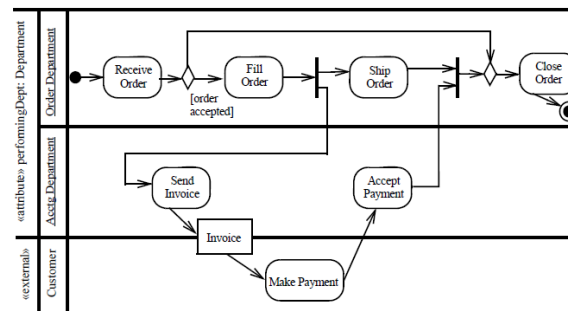
Tabel 2.11 Notasi pada Use Case Diagram

Notasi	Nama	Deskripsi
«extend» ←-----	<i>Extends</i>	Notasi hubungan antar <i>use case</i> yang menunjukkan bahwa <i>behavior</i> sebuah <i>use case</i> dapat diperluas (<i>extended</i>) oleh <i>use case</i> yang lain.
«include» -----→	<i>Include</i>	Notasi hubungan antar <i>use case</i> yang menunjukkan bahwa sebuah <i>behavior</i> dari sebuah <i>use case</i> merupakan bagian dari <i>use case</i> lain.

Sumber: Diadaptasi dari Object Management Group (2005)

2.7.2 Activity Diagram

Activity diagram digunakan untuk memodelkan aspek dinamis dari sebuah sistem (Booch, et al., 1998). *Activity diagram* pada umumnya digunakan untuk memodelkan urutan langkah dalam proses komputasi. *Activity diagram* menekankan pemodelan alur antar aktivitas, namun *activity diagram* juga dapat digunakan untuk memodelkan alur perubahan *state* dari sebuah objek. Contoh *activity diagram* terdapat dalam Gambar 2.2.



Gambar 2.2 Contoh Activity Diagram

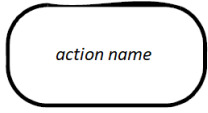
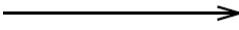
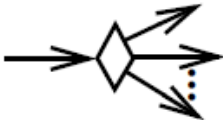
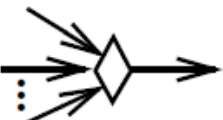
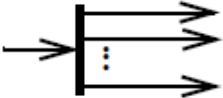
Sumber: Object Management Group (2005)

Activity diagram memiliki beberapa notasi, beberapa notasi yang digunakan dalam pemodelan di penelitian ini terdiri dari notasi *initial node*, notasi *actions*, notasi *activity edges*, notasi *decision node* atau *merge node*, notasi *fork node* atau *join node*, notasi *final node (activity final)*, dan notasi *activity partition*. Penjelasan setiap notasi pada activity diagram terdapat pada Tabel 2.12.

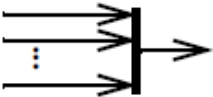


Tabel 2.12 Notasi pada Activity Diagram

Notasi	Nama	Deskripsi
●	<i>Initial Node</i>	Notasi kontrol yang menandakan titik awal untuk mengeksekusi sebuah aktivitas yang tergambar di dalam

Tabel 2.12 Notasi pada Activity Diagram

Notasi	Nama	Deskripsi
		<i>activity diagram</i> . Notasi <i>initial node</i> tidak boleh memiliki <i>incoming edges</i> .
	<i>Actions</i>	Notasi yang merepresentasikan tindakan yang dilakukan oleh sebuah unit di dalam sistem atau unit bisnis.
	<i>Activity Edges</i>	Notasi yang menghubungkan serangkaian notasi di dalam <i>activity diagram</i> .
	<i>Decision Node</i>	Notasi yang digunakan untuk menangani alur kondisional. Notasi ini memiliki 1 <i>incoming edges</i> dan memiliki 2 atau lebih <i>outgoing edges</i> . Masing-masing <i>outgoing edges</i> harus memiliki nilai boolean yang digambarkan melalui <i>guards</i> . Notasi ini tidak merepresentasikan proses sinkronisasi, sehingga alur akan berjalan melalui salah satu <i>outgoing edges</i> sesuai dengan kondisi yang dipenuhi.
	<i>Merge Node</i>	Notasi yang digunakan untuk menerima beberapa alur alternatif. Notasi ini tidak merepresentasikan proses sinkronisasi, melainkan menerima salah satu dari beberapa alur alternatif. Notasi ini memiliki 2 atau lebih <i>incoming edges</i> dan memiliki 1 <i>outgoing edges</i> .
	<i>Fork Node</i>	Notasi yang merepresentasikan pembagian sebuah alur menjadi beberapa alur yang berjalan secara <i>concurrent</i> . <i>Fork nodes</i> memiliki 1 <i>incoming edges</i> dan 2 atau lebih <i>outgoing edges</i> . <i>Outgoing edges</i> bisa saja memiliki <i>guards</i> untuk menandakan kondisi tertentu ketika alur melewati <i>edges</i> tersebut, hal ini disebut sebagai <i>conditional fork</i> . Jika <i>conditional fork</i> berlaku, maka alur

Tabel 2.12 Notasi pada Activity Diagram

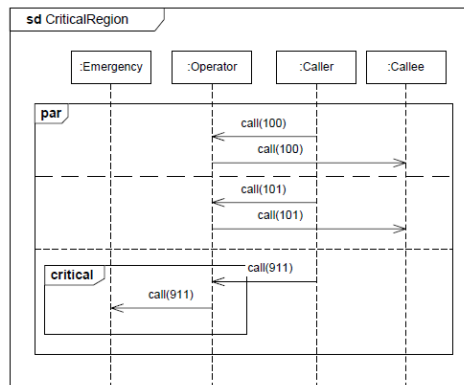
Notasi	Nama	Deskripsi
	<i>Join Node</i>	Notasi yang merepresentasikan proses sinkronisasi beberapa flow. Notasi ini memiliki 2 atau lebih <i>incoming edges</i> dan 1 <i>outgoing edges</i> .
	<i>Final Node (Activity Final)</i>	Notasi yang merepresentasikan titik berakhirnya alur aktivitas. <i>Final node</i> tidak boleh memiliki <i>outgoing edges</i> .
	<i>Activity Partition</i>	Notasi yang digunakan untuk mengelompokkan <i>actions</i> berdasarkan unit sistem, unit bisnis, atau entitas diluar sistem yang menjalankan <i>actions</i> tersebut. Penamaan notasi disesuaikan dengan nama unit sistem, nama unit bisnis, atau nama entitas yang melakukan <i>actions</i> tertentu.

Sumber: Diadaptasi dari Object Management Group (2005)

2.7.3 Sequence Diagram

Sequence diagram merupakan varian dari *interaction diagram* yang memvisualisasikan interaksi, yang terdiri dari sekumpulan objek dan hubungan diantaranya, termasuk pesan yang mungkin dikirimkan antar objek (Booch, et al., 1998). *Sequence diagram* menekankan visualisasi pertukaran pesan antar objek didalam sistem dalam urutan waktu. Contoh *sequence diagram* terdapat dalam Gambar 2.3.

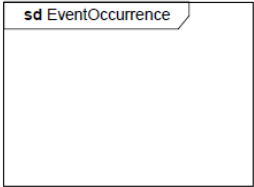
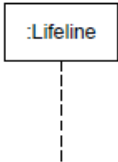
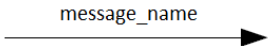
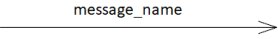
Sequence diagram berisi notasi-notasi yang merepresentasikan *instances* dari sebuah kelas, antarmuka, komponen, dan *nodes*, beserta pertukaran pesan di antara *instances* tersebut. Beberapa notasi *sequence diagram* yang digunakan pada penelitian ini dijelaskan pada Tabel 2.13.



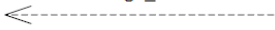
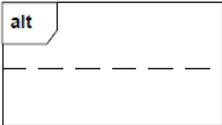
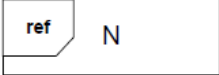
Gambar 2.3 Contoh Sequence Diagram

Sumber: Object Management Group (2005)

Tabel 2.13 Notasi pada Sequence Diagram

Notasi	Nama	Deskripsi
	<i>Frame</i>	Notasi yang berisi sekumpulan interaksi. Didalam notasi ini terdapat serangkaian <i>instances</i> yang saling bertukar pesan sebagai visualisasi aspek dinamis bagian dari sistem. Nama interaksi dituliskan di dalam bangun segi lima di pojok kiri atas notasi <i>frame</i> .
	<i>Lifeline</i>	Notasi ini merupakan representasi sebuah <i>instances</i> yang terlibat dalam sebuah interaksi pertukaran pesan (<i>message</i>).
	<i>Message (call)</i>	Notasi yang merepresentasikan komunikasi yang bersifat <i>synchronous</i> di antara <i>lifeline</i> . Notasi <i>call</i> secara khusus merepresentasikan pemanggilan operasi atau metode. Jika notasi ini digunakan untuk memodelkan pemanggilan operasi dengan argumen, maka argumen tersebut harus disertakan dalam penamaan notasi.
	<i>Message (asynchronous)</i>	Notasi yang merepresentasikan komunikasi yang bersifat <i>asynchronous</i> di antara <i>lifeline</i> .

Tabel 2.13 Notasi pada *Sequence Diagram*

Notasi	Nama	Deskripsi
	<i>Message (reply)</i>	Notasi yang merepresentasikan pesan balasan dari penggunaan notasi <i>message call</i> atau <i>asynchronous</i> .
	<i>Combined Fragment (Fragment)</i>	<p>Notasi yang digunakan untuk mengelompokkan interaksi kedalam fragmen. Setiap fragmen dapat memiliki beberapa <i>operand</i> yang masing-masing ditempatkan pada jalur yang dipisahkan oleh garis putus-putus di dalam fragmen. Masing-masing <i>operand</i> dapat memiliki sebuah <i>guard</i>, yang menandakan sebuah kondisi yang dipenuhi ketika <i>operand</i> tersebut dijalankan. Notasi ini memiliki beberapa varian yang ditandai dengan penamaan <i>interaction operator</i> di dalam bangun segi lima di pojok kiri atas notasi ini.</p> <p>Varian <i>alternative</i> (alt) merupakan varian yang merepresentasikan interaksi kondisional. Pada varian ini setidaknya 1 <i>operand</i> akan dijalankan.</p> <p>Varian <i>option</i> (opt) merupakan varian yang merepresentasikan interaksi kondisional. Varian ini hanya boleh memiliki 1 <i>operand</i>, yang bisa saja tidak dijalankan apabila kondisi pada <i>guard</i> tidak terpenuhi.</p> <p>Varian <i>loop</i> merupakan varian yang merepresentasikan interaksi yang dilakukan berulang-ulang selama kondisi <i>operand</i> terpenuhi. Varian ini hanya boleh memiliki 1 <i>operand</i>.</p>
	<i>Interaction Use (Fragment)</i>	Notasi yang digunakan untuk merujuk sebuah interaksi. Pemodelan notasi ini menggunakan

Tabel 2.13 Notasi pada *Sequence Diagram*

Notasi	Nama	Deskripsi
		notasi <i>combined fragment</i> dengan nama <i>interaction operator</i> ref. Nama interaksi yang dirujuk diletakkan di tengah notasi.

Sumber: Diadaptasi dari Object Management Group (2005)

2.7.4 *Class Diagram*

Class diagram menunjukkan sekumpulan kelas (*class*), *interfaces*, kolaborasi (*collaboration*) dan hubungan (*relationship*) di antaranya (Booch, et al., 1998). *Class diagram* umumnya berisi beberapa elemen yang terdiri dari kelas, *interfaces*, kolaborasi, *relationship* (*dependency*, *generalization*, dan *association*).

Class diagram digunakan untuk memodelkan aspek statis dari sebuah sistem (Booch, et al., 1998). Beberapa cara pemodelan aspek statis yang dapat dimodelkan ke dalam *class diagram* di antaranya yaitu:

1. *Modeling the Vocabulary of a System*

Pemodelan *class diagram* pada umumnya digunakan untuk memodelkan abstraksi dari sebuah sistem yang didapatkan dari masalah yang ingin diselesaikan dengan sistem yang akan dibangun. Masing-masing abstraksi merupakan bagian dari *vocabulary* dari sebuah sistem. Setiap abstraksi dimodelkan ke dalam kelas-kelas beserta atribut dan kemampuan yang dapat dilakukan oleh kelas-kelas tersebut.

2. *Modeling Simple Collaboration*

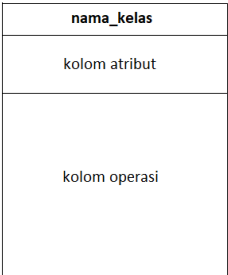
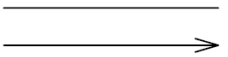



Kolaborasi dalam hal pemodelan *class diagram* merepresentasikan hubungan antar kelas, *intrefaces*, dan objek lain yang bekerja bersama-sama untuk menunjukkan *behavior* dari sebuah sistem.

3. *Modeling a Logical Database Schema*

Class diagram dapat digunakan untuk memodelkan skema basis data sebagai cetak biru konsep rancangan basis data. Skema yang dimodelkan merepresentasikan objek-objek yang akan menyimpan data yang akan dibutuhkan oleh sistem.

Notasi kelas memiliki struktur yang terdiri dari 3 kolom di antaranya yaitu kolom nama kelas, kolom atribut, dan kolom operasi. Masing-masing atribut dan operasi yang dimiliki oleh kelas dapat ditentukan visibilitasnya dan digambarkan dengan notasi *visibility kind* yang diletakkan sebelum nama atribut atau nama operasi. Penjelasan notasi *class diagram* yang digunakan pada penelitian ini terdapat pada Tabel 2.14.

Tabel 2.14 Notasi pada *Class Diagram*

Notasi	Nama	Deskripsi
	<i>Class</i>	<p>Notasi yang merepresentasikan sebuah objek beserta sekumpulan atribut dan operasi yang dimiliki. Struktur notasi ini memiliki 3 kolom yang terdiri dari kolom nama kelas, kolom atribut, dan kolom operasi.</p> <p>Masing-masing atribut dan operasi dapat ditentukan visibility-nya dengan</p>
	<i>Association</i>	Notasi yang merepresentasikan relasi sebuah objek dengan objek yang lain.
	<i>Generalization</i>	Notasi yang merepresentasikan relasi sebuah objek yang spesifik (<i>subclass</i>) dengan objek yang lebih umum (<i>superclass</i>). <i>Subclass</i> mewarisi atribut dan operasi yang dimiliki oleh <i>superclass</i> .
	<i>Aggregation</i>	Notasi yang merepresentasikan relasi sebuah objek yang lebih luas (<i>whole</i>) dengan objek yang menjadi bagian dari objek tersebut (<i>parts</i>). Objek <i>whole</i> diletakkan pada sisi ujung notasi yang memiliki bentuk belah ketupak, sedangkan objek <i>parts</i> diletakkan pada ujung yang lain.
	<i>Composition</i>	Notasi ini merupakan bentuk lain dari notasi <i>aggregation</i> , namun dalam notasi ini sebuah objek <i>parts</i> merupakan bagian dari hanya 1 objek yang lebih besar.
+	<i>Public</i> (<i>Visibility Kind</i>)	Notasi yang digunakan untuk menjelaskan aksesibilitas atribut atau operasi yang dimiliki objek. <i>Public</i> memungkinkan atribut atau operasi dapat diakses secara langsung oleh objek lain yang memiliki relasi dengan objek yang memilikinya.
#	<i>Protected</i> (<i>Visibility Kind</i>)	<i>Protected</i> memungkinkan atribut atau operasi dapat diakses secara langsung oleh <i>instance</i> dari objek <i>subclass</i> .

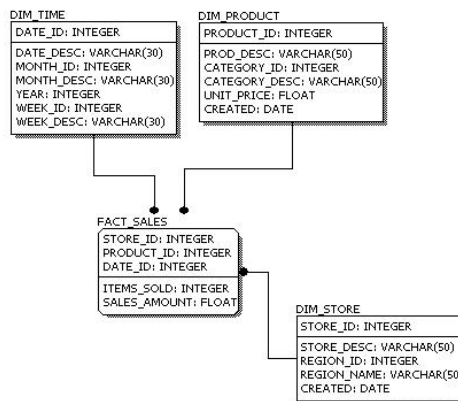
Tabel 2.14 Notasi pada *Class Diagram*

Notasi	Nama	Deskripsi
-	<i>Private</i> (<i>Visibility Kind</i>)	<i>Private</i> memungkinkan atribut atau operasi hanya dapat diakses oleh objek yang memilikinya.

Sumber: Diadaptasi dari Object Management Group (2005)

2.8 *Physical Data Model*

Physical Data Model (PDM) memiliki konsep yang sama dengan pemodelan kelas, namun tujuan PDM adalah untuk memodelkan skema internal basis data, menggambarkan tabel beserta kolom-kolomnya, dan hubungan antar tabel (Ambysoft Inc., 2013).



Gambar 2.4 Contoh *Physical Data Model*

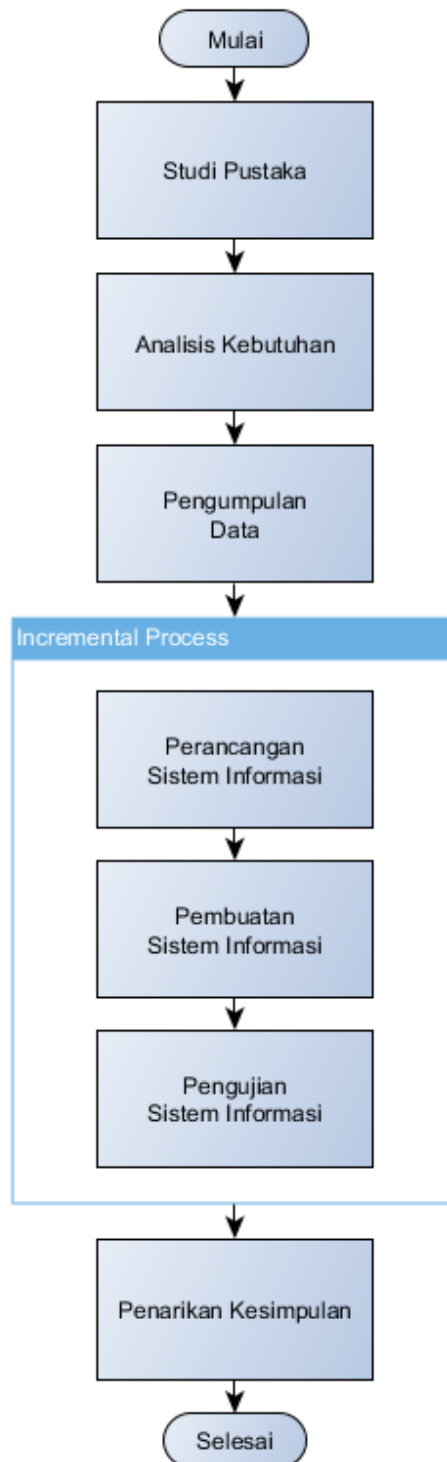
Sumber: 1keydata (2009)

Pemodelan data dalam PDM dapat dilakukan berdasarkan hasil pemodelan kelas dalam *class diagram*. Contoh PDM terdapat dalam Gambar 2.4. Booch, et al., menjelaskan, pemodelan PDM dapat dilakukan dengan 3 strategi, di antaranya yaitu:

1. Mendefinisikan masing kelas pada *class diagram* menjadi tabel pada PDM. Hal ini merupakan strategi yang paling sederhana, namun akan mendatangkan kesulitan ketika rancangan *class diagram* mengalami perubahan, misal, terdapat *subclass* yang ditambahkan.
2. Mengumpulkan *subclass* dengan *superclass* pada tingkat hirarki yang sama, kemudian definisikan masing-masing kelas menjadi tabel pada PDM. Kekurangan strategi ini adalah akan menyebabkan kapasitas penyimpanan cepat berkurang.
3. Memisahkan *subclass* dengan *superclass* dan mendefinisikannya menjadi tabel pada PDM. Kekurangan dari strategi ini adalah akan membutuhkan mekanisme *join table* untuk proses pengambilan data melalui basis data.

BAB 3 METODOLOGI

3.1 Alur Penelitian



Gambar 3.1 Alur Penelitian

3.1.1 Studi Pustaka

Dalam Gambar 3.1 penelitian dimulai dengan melakukan studi pustaka, yaitu dengan mengkaji penelitian terdahulu mengenai Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” yang menjadi landasan pengembangan lanjut SIKES “ANDAL”. Selain itu, pada tahap studi pustaka dilakukan pemahaman teori dan kerangka kerja yang berkaitan dan digunakan sebagai panduan untuk menyelesaikan penelitian, contohnya: teori tentang *Software Development Life Cycle* (SDLC), *Unified Modeling Language* (UML), dan perancangan arsitektur.

3.1.2 Analisis Kebutuhan

Analisis kebutuhan dilakukan dengan melakukan analisa terhadap proses bisnis yang ada pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”, selain itu dilakukan juga dengan melakukan wawancara untuk mendapatkan informasi mengenai siapa saja *stakeholder*, aktor, dan tipe pengguna Sistem Informasi Kesehatan “ANDAL”. Wawancara juga digunakan untuk lebih memahami permasalahan dan kemungkinan solusi yang akan digunakan, selain itu wawancara juga digunakan untuk memastikan efektifitas wawancara itu sendiri, agar responden atau narasumber merasa nyaman dalam memberikan jawaban dalam kegiatan wawancara.

Analisa kebutuhan memiliki tujuan agar peneliti dapat memahami permasalahan dengan baik, menentukan batasan masalah yang akan diselesaikan, mengetahui lebih jauh siapa saja *stakeholder*, aktor, dan calon pengguna SIKES “ANDAL”, mengetahui lingkungan sistem secara teknis, mengetahui kebutuhan atau fungsi-fungsi yang harus diimplementasikan pada sistem, serta kemungkinan solusi dan kebutuhan yang akan digunakan dan dipenuhi untuk menyelesaikan masalah. Dalam penelitian ini wawancara dilakukan kepada *stakeholder* SIKES “ANDAL” yang meliputi dr. Jack Roebijoso dan peneliti yang menerbitkan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”.

Kegiatan yang dilakukan dalam pengumpulan kebutuhan mengadaptasi beberapa tugas dalam proses *Requirement Engineering* (RE) khususnya *inception*, *elicitation*, dan *negotiation*, seperti yang dijelaskan oleh Pressman (2010). Proses RE digunakan sebagai panduan untuk membantu peneliti memeriksa bagaimana SIKES “ANDAL” seharusnya berjalan, memahami kebutuhan secara spesifik, dan menentukan prioritas kebutuhan yang harus diimplementasikan. Hasil kegiatan analisa kebutuhan diantaranya daftar aktivitas-aktivitas berdasarkan proses bisnis pada Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”, daftar fitur sistem informasi yang akan dibuat, serta kebutuhan fungsional dan non-fungsional sistem informasi yang akan dibuat. Hasil kegiatan analisa kebutuhan akan digunakan sebagai *input* pada tahap yang ada pada *incremental process*, yaitu perancangan sistem informasi.

3.1.3 Pengumpulan Data

Pengumpulan data dilakukan dengan mengumpulkan berkas-berkas data dasar yang berisi informasi identitas keluarga, riwayat penyakit atau kecelakaan yang

dialami anggota keluarga, serta perilaku kesehatan dan keselamatan melalui formulir data dasar yang diisi langsung oleh masyarakat pada daerah atau tempat kerja yang sudah ditentukan sebelumnya. Pengumpulan data akan dilakukan di Poliklinik Pabrik Gula Kebon Agung. Data yang sudah didapatkan akan disimpan kedalam basis data ketika tahap pembuatan sistem informasi.

3.1.4 Incremental Process

Kegiatan *Incremental process* dalam metodologi penelitian ini diadaptasi dari *Incremental Process Model* yang memiliki keuntungan, salah satunya, memungkinkan *delivery* dan *deployment* sistem kepada calon pengguna lebih cepat, meskipun belum semua fungsi dapat diimplementasikan kedalam sistem. Calon pengguna dapat dengan segera menggunakan dan mendapatkan nilai dari sistem yang di-*deliver* dibandingkan jika menerapkan model *Waterfall* (Sommerville, 2011). *Incremental process* pada penelitian ini mengadopsi pendekatan *plan-driven*, sehingga seluruh kebutuhan pengguna dan kebutuhan sistem sudah diidentifikasi terlebih dahulu pada tahap analisa kebutuhan.

Kemudian seluruh kebutuhan tersebut dapat dimodelkan sebagai panduan bagi peneliti untuk melakukan pembuatan perangkat lunak. Pembuatan perangkat lunak dilakukan dengan mengambil sebagian kebutuhan berdasarkan prioritasnya, kemudian hasil pengembangan tersebut dapat di-*deliver* kepada calon pengguna untuk dievaluasi dan diberi tanggapan. Hasil evaluasi dan tanggapan tersebut menjadi *input* bagi pengembangan pada *increment* selanjutnya hingga perangkat lunak dapat memberikan seluruh fitur yang dapat mencakup seluruh kebutuhan calon pengguna.

3.1.4.1 Perancangan Sistem Informasi

Perancangan sistem informasi meliputi pemodelan sistem kedalam notasi-notasi pada *Unified Modeling Language* (UML) berdasarkan hasil tahap analisa kebutuhan dan pengumpulan data. Seperti yang dijelaskan oleh Sommerville (2011) bahwa pemodelan digunakan untuk memperoleh kebutuhan sistem, menjelaskan kebutuhan sistem untuk memudahkan proses pengembangan sistem, serta untuk mendokumentasikan struktur dan operasi sistem. Hasil pemodelan sistem dapat membantu *stakeholder* memahami kebutuhan. Selain itu, bagi peneliti, hasil pemodelan digunakan sebagai dasar perancangan sistem dan sebagai dokumentasi dari sistem yang akan dibuat.

Pada penelitian ini, perancangan sistem informasi akan memanfaatkan *Use Case Diagram* untuk menunjukkan interaksi antara sistem dengan lingkungannya, *Activity Diagram* untuk menunjukkan aktivitas-aktivitas yang terlibat dalam sebuah proses dan pemrosesan data dalam sistem, *Sequence Diagram* untuk menunjukkan interaksi antara aktor dengan sistem dan antara aktor dengan komponen sistem, *Class Diagram* untuk menunjukkan kelas-kelas objek didalam sistem dan hubungan antar kelas tersebut.

3.1.4.2 Pembuatan Sistem Informasi

Pembuatan sistem informasi merupakan proses pengembangan SIKES “ANDAL” berdasarkan hasil perancangan pada tahap perancangan sistem informasi. SIKES “ANDAL” dibangun menggunakan bahasa pemrograman PHP, dengan framework Codeigniter. SIKES “ANDAL” dari penelitian ini merupakan purwarupa dengan fitur inti berdasarkan Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”. Pembuatan sistem informasi dilakukan sebagai bagian dari *incremental process model*, sehingga pembuatan sistem informasi akan dilakukan secara berkala dengan mengambil beberapa bagian dari kebutuhan pengguna dan kebutuhan sistem untuk diimplementasikan pada setiap *increment*. Hasil dari setiap *increment* akan digunakan sebagai *input* pada tahap pengujian sistem informasi.

3.1.4.3 Pengujian Sistem Informasi

Pengujian merupakan sebuah elemen penting dari penjaminan kualitas perangkat lunak dan merepresentasikan *review* akhir dari spesifikasi, perancangan dan implementasi (Pressman, 2010). Pengujian SIKES “ANDAL” dilakukan dengan menggunakan pengujian *functional testing* untuk memastikan sistem dapat memberikan fungsi sesuai dengan yang telah ditentukan, serta untuk memastikan sistem tidak mengalami kegagalan pada saat digunakan.

Pengujian dilakukan terhadap hasil pembuatan sistem informasi pada setiap *increment*, jika pengujian dinyatakan tidak valid, maka pembuatan sistem pada *increment* selanjutnya digunakan untuk memperbaiki kesalahan yang telah dibuktikan melalui tahap pengujian. Jika pengujian dinyatakan valid dan masih terdapat kebutuhan sistem yang belum terimplementasi, maka *increment* selanjutnya digunakan untuk mengimplementasi kebutuhan-kebutuhan yang lain. Jika pengujian valid dan seluruh kebutuhan telah terimplementasi, maka penelitian dapat dianggap selesai. Hasil pengujian dari *increment* terakhir akan digunakan untuk tahap penarikan kesimpulan untuk menjawab rumusan masalah.

3.1.5 Penarikan Kesimpulan

Penarikan kesimpulan dilakukan berdasarkan penilaian terhadap hasil akhir akhir penelitian untuk menentukan apakah hasil penelitian dapat menjawab masalah yang telah dirumuskan.

BAB 4 ANALISIS PERSYARATAN

Sistem Informasi Kesehatan “ANDAL” saat ini telah dikembangkan dan mampu memberikan fitur pengelolaan data dasar kesehatan keluarga, pengelolaan data registrasi dan status pasien, menampilkan tingkat resiko kesehatan dan tingkat stres, pengelolaan hasil diagnosis pasien, sehingga sistem informasi kesehatan saat ini dapat membantu petugas klinik mengelola data dasar kesehatan keluarga, mengelola data registrasi dan status pasien, menampilkan tingkat resiko kesehatan dan tingkat stres, dan mengelola hasil diagnosis pasien.

Berdasarkan hasil wawancara dengan dr. Jack Roebijoso, sistem informasi kesehatan diharapkan dapat dikembangkan untuk dapat membantu petugas klinik mengelola data tindakan medis yang diberikan kepada pasien, mengelola data rujukan pasien, mengelola data pengecekan darah pasien, mengelola data kegiatan edukasi, advokasi, dan pengelolaan lingkungan tempat tinggal atau tempat kerja pasien dan mampu menampilkan persentase penyakit yang tersebar mulai dari tingkat RT, RW, kelurahan, kecamatan, hingga kota/kabupaten.

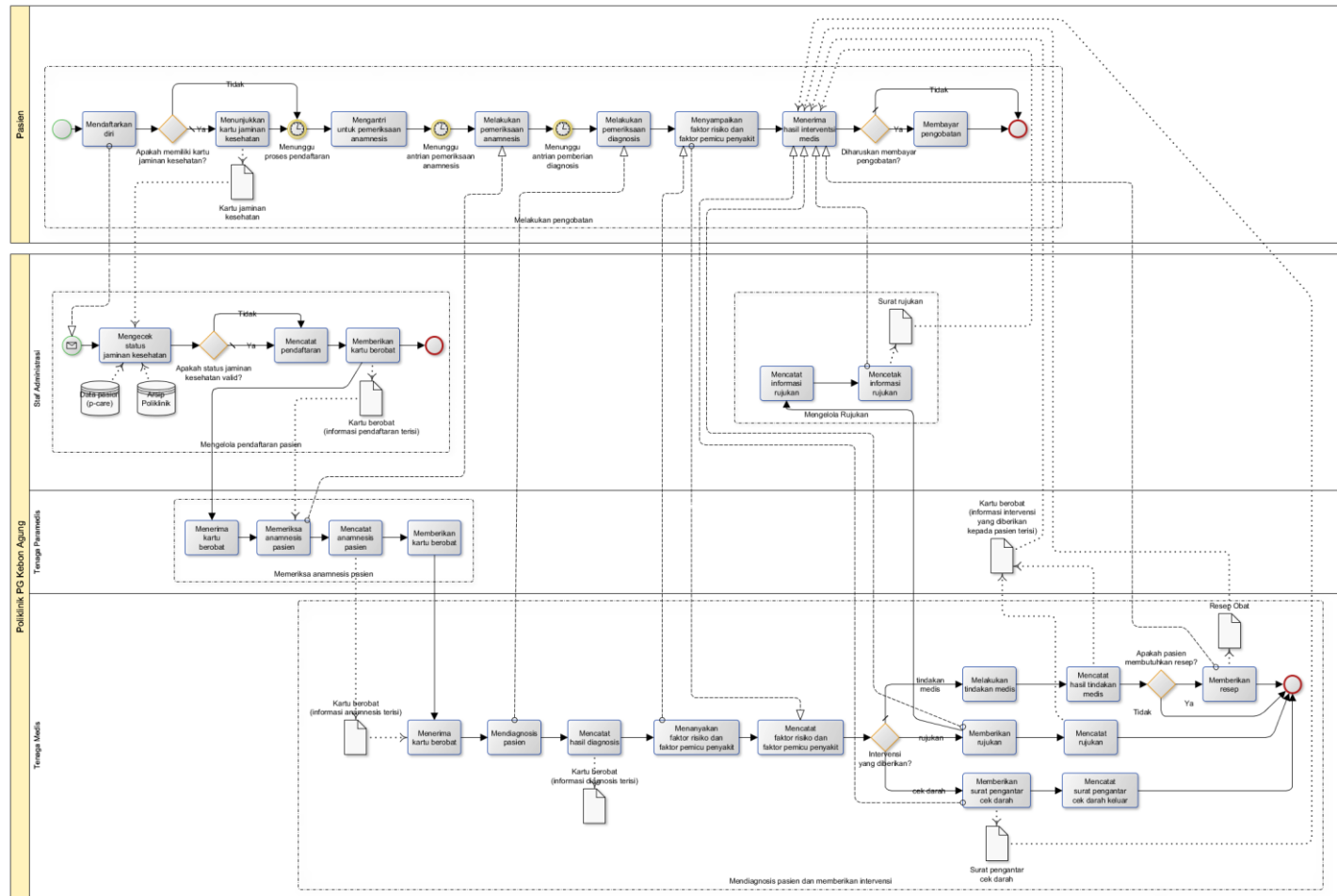
Bab ini membahas analisis persyaratan yang terdiri dari hasil analisis kebutuhan, fitur-fitur yang akan dikembangkan, persyaratan fungsional dan non-fungsional, perancangan *use case diagram* beserta *use case specification*, dan *activity diagram*.

4.1 Pemodelan Proses Bisnis

Pemodelan proses bisnis dilakukan untuk memodelkan serangkaian aktivitas yang dilakukan oleh fungsi-fungsi bisnis pada Poliklinik Pabrik Gula Kebon Agung untuk memenuhi kebutuhan pengguna bisnis (internal maupun eksternal). Penggalan informasi mengenai proses bisnis yang akan dimodelkan didapatkan dengan cara melakukan wawancara dengan pegawai poliklinik mengenai alur proses bisnis pada poliklinik dan observasi pada lokasi studi kasus penelitian. Dalam penelitian ini, pemodelan proses bisnis akan menghasilkan beberapa diagram proses bisnis yang terbagi menjadi 2 kelompok, yaitu proses bisnis yang saat ini berjalan (*as-is*) dan proses bisnis usulan (*to-be*).

4.1.1 Proses Bisnis *As-Is*

Dalam penelitian ini proses bisnis *as-is* dapat diidentifikasi berdasarkan keadaan proses bisnis yang berjalan pada lokasi studi kasus sebelum solusi yang diajukan dalam penelitian ini diusulkan pada lokasi studi kasus. Analisis dan pemodelan terhadap proses bisnis *as-is* dilakukan agar peneliti dapat mengidentifikasi kemungkinan perbaikan atau peningkatan terhadap proses bisnis melalui solusi yang akan ditawarkan dalam penelitian ini. Analisis proses bisnis *as-is* dilakukan dengan melakukan wawancara dengan pegawai poliklinik mengenai alur aktivitas bisnis pada poliklinik dan observasi pada lokasi studi kasus penelitian. Kemudian hasil analisis dimodelkan dengan notasi BPMN untuk membantu pemangku kepentingan memahami hasil analisis proses bisnis *as-is*.

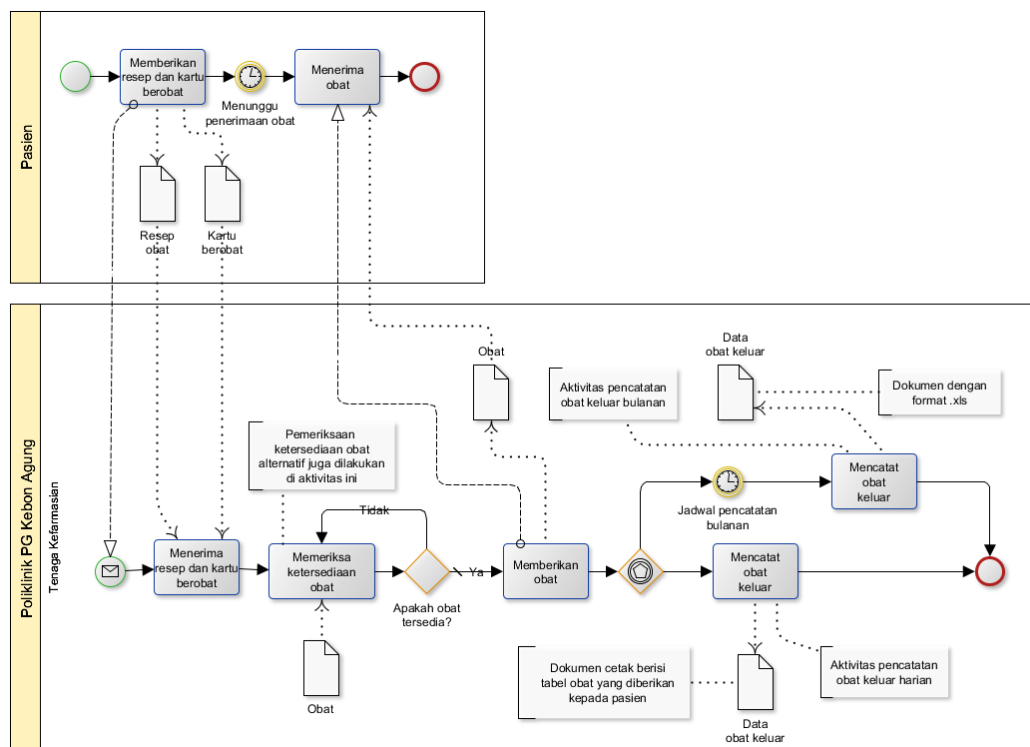


Gambar 4.1 Proses Bisnis As-Is Pelayanan Kesehatan

4.1.1.1 Proses Bisnis As-Is Pelayanan Kesehatan

Dalam Gambar 4.1 menjelaskan rangkaian aktivitas pelayanan kesehatan yang dilakukan oleh fungsi bisnis staf administrasi, tenaga paramedis, dan tenaga medis di Poliklinik Pabrik Gula Kebon Agung. Proses bisnis dimulai ketika Pasien sebagai pelanggan bisnis eksternal poliklinik ingin melakukan pengobatan dengan memulai aktivitas mendaftarkan diri melalui staf administrasi pada Poliklinik Pabrik Gula Kebon Agung. Kemudian staf administrasi melakukan serangkaian aktivitas pengelolaan pendaftaran pasien. Setelah staf administrasi memberikan kartu berobat kepada tenaga paramedis, tenaga paramedis akan melakukan serangkaian aktivitas pemeriksaan anamnesis pasien. Proses bisnis berlanjut ketika tenaga medis melakukan diagnosis terhadap pasien dan memberikan intervensi yang dibutuhkan pasien. Jika pasien membutuhkan rujukan, maka tenaga medis akan memberikan rujukan kepada pasien dan proses pengelolaan rujukan akan dilanjutkan oleh staf administrasi. Setelah diagnosis dan intervensi telah diberikan kepada pasien, pasien akan menerima informasi hasil intervensi yang akan diberikan oleh tenaga medis. Pasien dapat melanjutkan proses pembayaran atau menyelesaikan aktivitas pengobatan di poliklinik.

4.1.1.2 Proses Bisnis As-Is Pencatatan Obat Keluar



Gambar 4.2 Proses Bisnis As-Is Pencatatan Obat Keluar

Proses bisnis *as-is* pencatatan obat keluar merupakan serangkaian aktivitas yang memenuhi kebutuhan pasien ketika harus menerima obat sesuai resep dokter. Analisis proses bisnis tersebut dilakukan dengan melakukan wawancara dengan tenaga kefarmasian Poliklinik Pabrik Gula Kebon Agung mengenai alur

aktivitas yang dilakukan ketika pasien membutuhkan obat sesuai resep dokter hingga pasien menerima obat tersebut. Bagi tenaga kefarmasian, proses bisnis *as-is* pencatatan obat keluar berisi serangkaian aktivitas pemeriksaan ketersediaan obat yang dilakukan setelah tenaga kefarmasian menerima resep dokter dan kartu berobat dari pasien. Selain itu, terdapat aktivitas pencatatan obat keluar yang dilakukan setiap hari dan setiap bulan.

Pencatatan obat keluar harian dilakukan setelah obat sesuai resep dokter diberikan kepada pasien yang membutuhkan. Pencatatan obat keluar harian dan bulanan masing-masing disimpan dalam dokumen yang berbeda. Pencatatan obat keluar harian disimpan dalam dokumen cetak berisi tabel obat yang tersedia di poliklinik dan beberapa informasi mengenai pencatatan obat keluar harian, sedangkan dokumen pencatatan obat keluar bulanan disimpan dalam bentuk dokumen yang dapat dikelola melalui aplikasi Microsoft Excel.

4.1.2 Proses Bisnis *To-be*

Proses bisnis *to-be* yang dimodelkan pada penelitian ini merupakan proses bisnis usulan yang ditawarkan dalam penelitian ini kepada pemangku kepentingan. Proses bisnis *to-be* diidentifikasi berdasarkan hasil analisis terhadap proses bisnis *as-is*. Dalam penelitian ini, analisis proses bisnis *to-be* dilakukan dengan melakukan eliminasi pada aktivitas yang bersifat redundan, kemudian menentukan aktivitas-aktivitas yang membutuhkan dukungan solusi sistem informasi. Selanjutnya, proses bisnis *to-be* dimodelkan untuk menggambarkan perubahan atau tambahan yang diusulkan melalui penelitian ini. Pada penelitian ini, hasil pemodelan proses bisnis *to-be* digunakan sebagai *input* kegiatan analisis persyaratan.

4.1.2.1 Proses Bisnis *To-Be* Pelayanan Kesehatan

Proses bisnis *to-be* pelayanan kesehatan berisi proses bisnis usulan terkait proses pelayanan kesehatan pada Poliklinik Pabrik Gula Kebon Agung. Solusi yang tergambar dalam model proses bisnis *to-be* adalah pemanfaatan lokasi penyimpanan data terpusat yang dapat dikelola oleh beberapa fungsi bisnis poliklinik.

Lokasi penyimpanan data yang diusulkan dapat digunakan untuk mengelola data pendaftaran pengobatan pasien, data anamnesis pasien, data hasil diagnosis pasien, data faktor risiko dan pemicu penyakit, data tindakan medis yang diberikan kepada pasien, data rujukan, dan data surat pengantar cek darah yang dikeluarkan untuk pasien.

Sehingga beberapa aktivitas pada proses bisnis *to-be* pelayanan kesehatan dapat dieliminasi, beberapa aktivitas yang dieliminasi dari proses bisnis *as-is* di antaranya yaitu:

1. Aktivitas staf administrasi untuk memberikan kartu berobat kepada tenaga paramedis setelah aktivitas mencatat pendaftaran dan aktivitas tenaga paramedis untuk menerima kartu berobat sebelum melakukan aktivitas memeriksa anamnesis pasien.

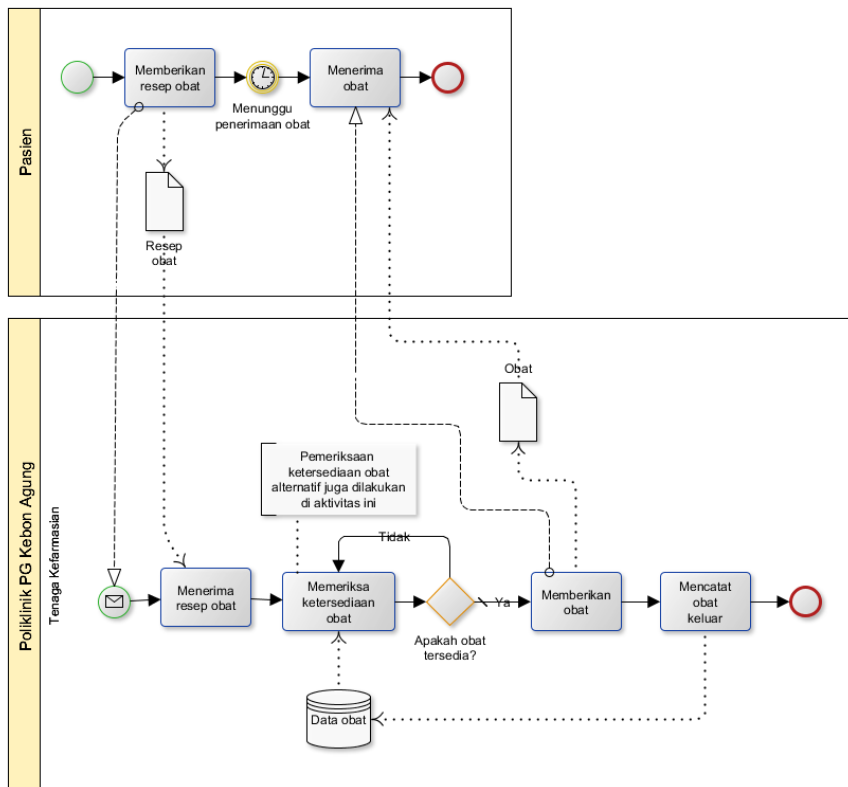
2. Aktivitas tenaga paramedis untuk memberikan kartu berobat kepada tenaga medis setelah aktivitas mencatat anamnesis pasien dan aktivitas tenaga medis untuk menerima kartu berobat sebelum melakukan aktivitas mendiagnosis pasien.

Pada proses bisnis *to-be* pelayanan kesehatan terdapat penggunaan data pada aktivitas mencatat hasil diagnosis dan aktivitas menanyakan faktor risiko dan faktor pemicu penyakit yang dilakukan oleh tenaga medis. Data tersebut masing-masing disimpan pada lokasi penyimpanan data modul diagnosis penyakit dan lokasi penyimpanan data modul faktor risiko dan faktor pemicu penyakit. Model proses bisnis *to-be* terdapat dalam Gambar 4.4.

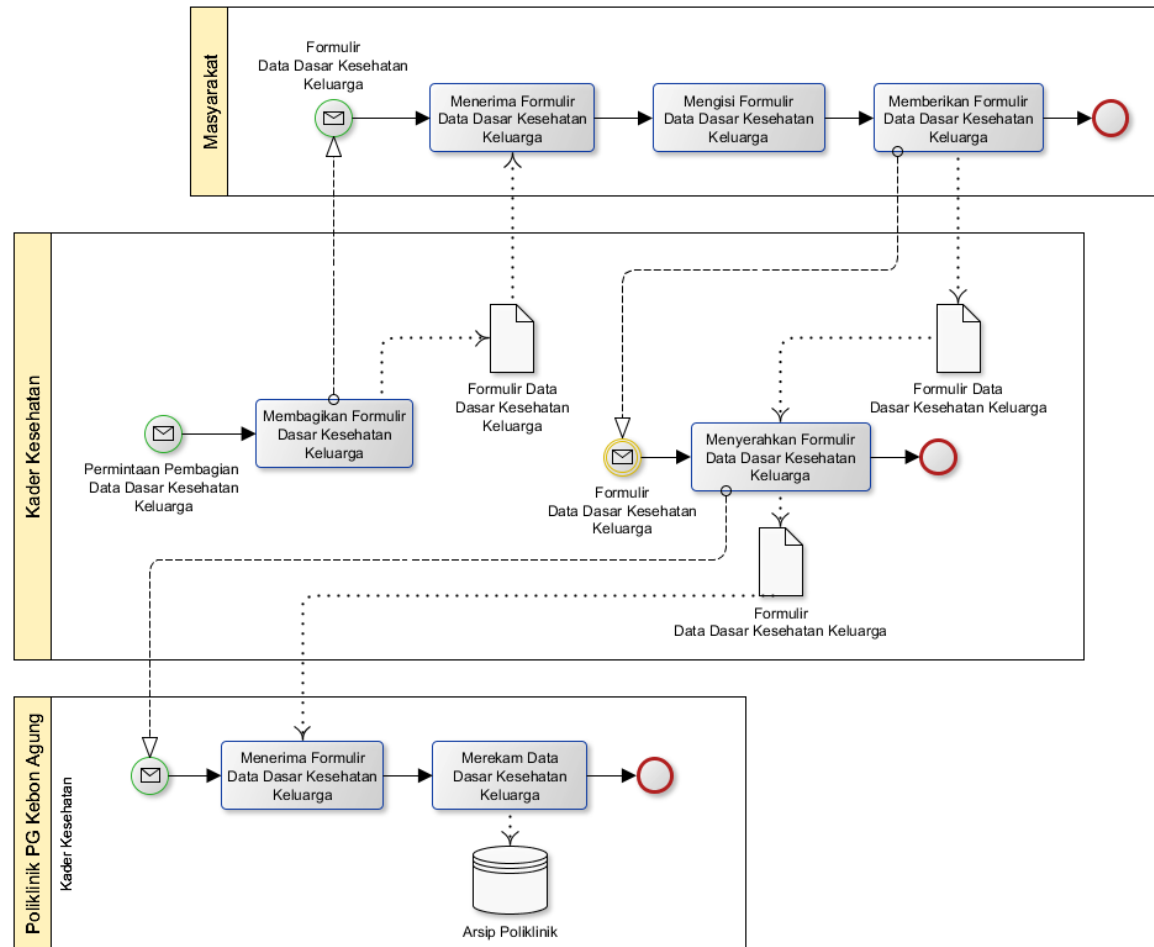
4.1.2.2 Proses Bisnis To-Be Pencatatan Obat Keluar

Proses bisnis *to-be* pencatatan obat keluar merupakan bisnis usulan terkait kegiatan pencatatan obat keluar pada Poliklinik Pabrik Gula Kebon Agung. Solusi yang ditawarkan untuk proses pencatatan obat keluar adalah penggunaan lokasi penyimpanan data terpusat untuk kebutuhan pengelolaan data obat.

Solusi yang ditawarkan dapat mengkombinasikan aktivitas pencatatan obat keluar yang pada proses bisnis *as-is* harus dilakukan secara rutin setiap hari dan setiap bulan. Pada proses bisnis *to-be*, aktivitas pencatatan obat keluar dapat dilakukan setiap aktivitas memberikan obat kepada pasien dilakukan tanpa harus mengulangi aktivitas pencatatan obat keluar setiap bulan. Model proses bisnis *to-be* pencatatan obat keluar terdapat dalam Gambar 4.3.



Gambar 4.3 Proses Bisnis To-Be Pencatatan Obat Keluar



Gambar 4.5 Proses Bisnis To-Be Pencatatan Data Dasar Kesehatan Keluarga

4.1.2.3 Proses Bisnis To-Be Pencatatan Data Dasar Kesehatan Keluarga

Proses bisnis ini merupakan usulan berdasarkan kebutuhan rencana penerapan Model Organisasi dan Manajemen “ANDAL”. Proses pencatatan data dasar kesehatan pasien merupakan proses pengumpulan data pasien yang dikelompokkan berdasarkan data keluarga pasien. Proses pencatatan data dasar kesehatan pasien memanfaatkan formulir dengan format cetak.

Secara umum informasi yang dikumpulkan melalui proses ini adalah informasi mengenai identitas pasien, perkawinan kepala keluarga, anggota keluarga, keadaan ekonomi keluarga, perilaku kesehatan dan keselamatan keluarga, riwayat kesehatan keluarga, dan variabel-variabel untuk menentukan tingkat stres kepala keluarga.

Informasi mengenai alur proses diperoleh berdasarkan hasil wawancara dengan dr. Jack Roebijoso selaku kepala Poliklinik Pabrik Gula Kebon Agung dan penggagas Model Organisasi dan Manajemen “ANDAL”. Model proses bisnis to-be pencatatan data dasar kesehatan keluarga terdapat dalam Gambar 4.5.

4.2 Analisis Persyaratan

4.2.1 Identifikasi Tipe Pemangku Kepentingan

Identifikasi tipe pemangku kepentingan dilakukan untuk mengetahui dan mengelompokkan beberapa pemangku kepentingan sesuai dengan karakteristik pemangku kepentingan dan hubungan pemangku kepentingan dengan sistem yang akan dikembangkan dalam penelitian ini. Informasi mengenai pemangku kepentingan diperoleh dengan melakukan wawancara dengan Kepala Poliklinik Pabrik Gula Kebon Agung sekaligus penggagas Model Organisasi dan Manajemen Kesehatan Primer “ANDAL”, dr. Jack Roebijoso.

Hasil identifikasi tipe pemangku kepentingan adalah daftar contoh pemangku kepentingan yang telah dikelompokkan sesuai karakteristik dan hubungan pemangku kepentingan terhadap sistem yang akan dikembangkan dalam penelitian ini. Hasil identifikasi tipe pemangku kepentingan digunakan sebagai informasi untuk langkah analisis persyaratan berikutnya, yaitu analisis permasalahan, sehingga peneliti dapat memperoleh informasi masalah dari berbagai pemangku kepentingan yang ada pada lokasi studi kasus. Penjelasan tipe pemangku kepentingan untuk penelitian ini terdapat pada Tabel 4.1.

Tabel 4.1 Tipe Pemangku Kepentingan

Tipe Stakeholder	Deskripsi	Contoh Stakeholder
Pengguna	Individu yang berinteraksi dengan sistem informasi kesehatan secara langsung dengan berperan sebagai aktor pada pemodelan <i>use case</i> .	Petugas Poliklinik Pabrik Gula Kebon Agung yang terdiri dari Kepala Poliklinik, Dokter, Perawat, Petugas

Tabel 4.1 Tipe Pemangku Kepentingan (lanjutan)

Tipe Stakeholder	Deskripsi	Contoh Stakeholder
		Farmasi, Petugas Administrasi.
Pengembang	Organisasi atau individu yang melakukan pengembangan sistem informasi kesehatan.	Peneliti.
Pihak yang berwenang	Organisasi atau individu yang memiliki kemampuan untuk mendukung pengembangan solusi dengan cara memberikan informasi mengenai regulasi yang berjalan, sehingga solusi yang akan dibangun dapat berjalan sesuai dengan aturan yang berlaku..	Poliklinik Pabrik Gula Kebon Agung.
Pelanggan	Organisasi atau individu yang akan mendapatkan manfaat dari sistem informasi kesehatan.	Poliklinik Pabrik Gula Kebon Agung, pasien Poliklinik Pabrik Gula Kebon Agung.

4.2.2 Analisis Masalah

Analisis masalah dilakukan untuk membantu peneliti memahami masalah yang terjadi dan harus diselesaikan oleh pemangku kepentingan. Dalam penelitian ini, analisis masalah dilakukan dengan melakukan wawancara dengan petugas administrasi dan apoteker Poliklinik Pabrik Gula Kebon Agung. Problem statement yang didapatkan dari wawancara petugas administrasi dan apoteker Poliklinik Pabrik Gula Kebon Agung terdapat pada Tabel 4.2.

Tabel 4.2 Problem Statement 1

The problem of	<ul style="list-style-type: none"> • Pencatatan data obat keluar yang harus dilakukan 2 kali kedalam 2 dokumen yang berbeda. • Rekapitulasi yang dilakukan dari 2 dokumen pencatatan obat keluar sering menunjukkan hasil yang tidak konsisten. • Hasil rekapitulasi data persediaan obat yang tidak konsisten menyulitkan petugas farmasi dalam membuat laporan persediaan obat.
Affects	Pengguna, pihak yang berwenang, dan pelanggan.

Tabel 4.2 Problem Statement 1

<i>The impact of which is</i>	Jumlah pemenuhan persediaan obat sering kali tidak sesuai dengan yang dibutuhkan.
<i>A successful solution would</i>	Sistem yang menyediakan layanan pengelolaan data obat dan layanan yang dapat digunakan untuk menghasilkan laporan persediaan obat.

Selain itu juga dilakukan wawancara dengan dr. Jack Roebijoso selaku penggagas Model Organisasi dan Manajemen Kesehatan Primer “ANDAL” dan Kepala Poliklinik Pabrik Gula Kebon Agung. Hasil analisis masalah adalah *problem statements* yang akan digunakan sebagai informasi untuk mengidentifikasi daftar kebutuhan pemangku kepentingan. *Problem statement* pada Tabel 4.3 merupakan hasil analisis masalah yang diperoleh berdasarkan wawancara dengan dr. Jack Roebijoso.

Tabel 4.3 Problem Statement 2

<i>The problem of</i>	<ul style="list-style-type: none">• Pencatatan data kesehatan yang tidak konsisten yang disebabkan duplikasi dan hilangnya dokumen cetak data kesehatan.• Penanganan kesehatan yang diberikan kepada pasien belum konsisten dan efektif karena tidak dilakukan berdasarkan data kesehatan yang konsisten.• Penanganan kesehatan yang diberikan kepada pasien belum konsisten dan efektif karena tidak ada panduan standar pelayanan kesehatan.
<i>Affects</i>	Pengguna, pihak yang berwenang, dan pelanggan.
<i>The impact of which is</i>	<ul style="list-style-type: none">• Perkembangan kesehatan pasien sulit diawasi.• Pencegahan penyakit terhadap pasien menjadi terhambat.• Kesulitan menemukan indikasi penyakit menular dan penyakit keturunan pada pasien.• Kualitas pelayanan kesehatan yang diberikan poliklinik kurang memuaskan.• Angka sakit dan angka kunjungan pasien ke poliklinik meningkat.
<i>A successful solution would</i>	Sistem yang menyediakan layanan pengelolaan data kesehatan terpusat dalam lingkup Poliklinik Pabrik Gula Kebon Agung.

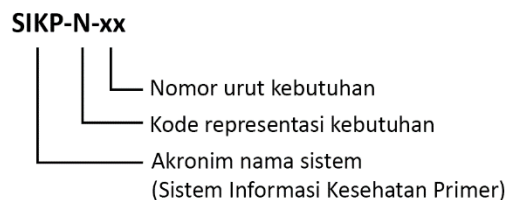
Pada Tabel 4.2 dan Tabel 4.3 merupakan informasi mengenai masalah yang dihadapi oleh pemangku kepentingan, siapa saja yang terkena dampak dari masalah yang teridentifikasi, informasi mengenai bagaimana dampak yang

dirasakan dari masalah yang teridentifikasi, dan solusi yang ditawarkan untuk menyelesaikan masalah.

4.2.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Pada penelitian ini, informasi mengenai kebutuhan pengguna diperoleh dengan melakukan analisis terhadap *problem statement* dan wawancara dengan Kepala Poliklinik Pabrik Gula Kebon Agung, petugas administrasi, dan petugas farmasi poliklinik. Kebutuhan pengguna berisi sekumpulan pernyataan yang berhubungan dengan masalah yang telah diidentifikasi pada tahap analisis masalah.

Hasil identifikasi kebutuhan pengguna digunakan sebagai dasar informasi tahap analisis fitur, yang merupakan tahap untuk mengidentifikasi solusi yang ditawarkan untuk memenuhi kebutuhan pengguna. Masing-masing kebutuhan pengguna diberi kode sebagai identitas, keterangan kodifikasi kebutuhan pengguna terdapat dalam Gambar 4.6.



Gambar 4.6 Keterangan Kodifikasi Kebutuhan Pengguna

Pada Tabel 4.4 berisi informasi kode identitas untuk masing-masing kebutuhan pengguna, pernyataan kebutuhan pengguna, pemangku kepentingan yang termasuk dalam kategori pengguna dan pengembang, yang memiliki keterkaitan dengan kebutuhan yang telah diidentifikasi, situasi saat ini yang berkaitan dengan kegiatan yang dilakukan untuk memenuhi kebutuhan pengguna, dan solusi yang ditawarkan untuk memenuhi masing-masing kebutuhan yang telah teridentifikasi.

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
SIKP-N-1	Sistem sebaiknya menyediakan layanan untuk pengelolaan data dasar kesehatan keluarga.	Petugas administrasi, perawat, dokter.	Pengumpulan dan pembaruan data dilakukan dengan mengisi formulir data dasar kesehatan keluarga. Data	Sistem informasi yang menyediakan layanan pencatatan data dasar kesehatan keluarga. Data disimpan secara historis,

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
			disimpan dalam bentuk dokumen cetak.	di dalam sebuah basis data.
SIKP-N-2	Sistem sebaiknya menyediakan layanan untuk pengelolaan data pengobatan pasien.	Petugas administrasi, perawat, dokter.	Data pengobatan pasien tersimpan dalam dokumen cetak yang disebut Kartu Berobat. Seluruh kartu berobat disimpan pada ruang arsip poliklinik. Data pada kartu berobat diisi apabila pasien pemilik kartu berobat datang untuk berobat.	Sistem informasi yang menyediakan layanan pencatatan dan penyimpanan data pengobatan pasien
SIKP-N-3	Sistem sebaiknya menyediakan layanan untuk penghitungan tingkat risiko penyakit dan tingkat stres berdasarkan data dasar kesehatan keluarga.	Dokter.	Tingkat risiko penyakit dan tingkat stres dihitung secara manual berdasarkan hasil pengisian formulir data dasar kesehatan keluarga.	Sistem informasi yang memberikan layanan penghitungan tingkat risiko penyakit dan tingkat stres. Tingkat risiko penyakit akan diperbarui oleh sistem apabila terdapat perubahan atau pengisian data dasar

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
				kesehatan keluarga.
SIKP-N-4	Sistem sebaiknya menyediakan layanan untuk pengelolaan data hasil diagnosis pasien dan intervensi kesehatan.	Dokter.	Data hasil diagnosis disimpan di dalam Kartu Berobat dan diisi ketika dokter memberikan diagnosis kepada pasien. Data intervensi kesehatan berupa tindakan medis disimpan di dalam Kartu Berobat. Data intervensi lain seperti surat rujukan dan surat pengantar cek darah masing-masing disimpan dalam format cetak.	Sistem informasi yang memberikan layanan pencatatan hasil diagnosis dan intervensi kesehatan yang diberikan kepada pasien. Data disimpan di dalam sebuah basis data yang dapat diakses oleh sistem informasi.
SIKP-N-5	Sistem sebaiknya menyediakan layanan pencatatan faktor risiko penyakit dan faktor pemicu penyakit pasien.	Dokter.	Tidak ada.	Sistem informasi yang memberikan layanan pencatatan faktor risiko penyakit pasien dan faktor pemicu penyakit pasien. Data faktor pemicu

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
				dan faktor risiko diisi ketika dokter menangani pasien di dalam ruang berobat, kemudian data disimpan di dalam sebuah basis data yang dapat diakses oleh sistem informasi.
SIKP-N-6	Sistem sebaiknya menyediakan layanan penyajian modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit.	Dokter, kepala poliklinik.	Modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit disimpan dalam format dokumen dan hanya dapat dibaca melalui aplikasi Microsoft Word.	Modul akan disimpan dalam format dokumen PDF, yang akan disimpan dalam satu lokasi penyimpanan yang dapat diakses oleh sistem informasi. Sistem informasi akan memberikan layanan untuk menampilkan modul yang tersimpan melalui aplikasi peramban (<i>browser</i>).
SIKP-N-7	Sistem sebaiknya menyediakan layanan	Petugas administrasi, dokter.	Surat rujukan tersimpan dalam format dokumen	Sistem informasi yang menyediakan layanan

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
	pencatatan surat rujukan yang dikeluarkan untuk pasien.		cetak. Salinan dokumen cetak disimpan di dalam arsip poliklinik. Surat rujukan diterbitkan melalui aplikasi Primary Care BPJS.	pencatatan salinan surat rujukan yang diberikan kepada pasien.
SIKP-N-8	Sistem sebaiknya menyediakan layanan pencatatan surat pengantar cek darah yang dikeluarkan untuk pasien.	Petugas administrasi, dokter.	Surat pengantar cek darah tersimpan dalam format dokumen cetak dan disimpan di dalam arsip poliklinik.	Sistem yang menyediakan layanan pencatatan surat pengantar cek darah yang diberikan kepada pasien
SIKP-N-9	Sistem sebaiknya menyediakan layanan pencatatan obat keluar.	Petugas farmasi.	Pencatatan obat dilakukan setiap hari dan setiap bulan. Data obat keluar harian dicatat dan disimpan dalam format dokumen cetak, sedangkan data obat keluar bulanan disimpan dalam format dokumen aplikasi Microsoft Excel.	Sistem informasi yang menyediakan layanan pencatatan obat keluar. Data obat keluar disimpan di dalam sebuah basis data yang dapat diakses melalui sistem informasi.

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
SIKP-N-10	Sistem sebaiknya menyediakan layanan yang dapat membantu penyediaan laporan data pencatatan obat.	Petugas farmasi.	Pelaporan dilakukan dengan melakukan pencocokan dan rekapitulasi data obat keluar harian dan data obat keluar bulanan. Data tersebut digunakan untuk proses pemenuhan persediaan obat poliklinik.	Sistem informasi yang menyediakan layanan untuk membantu pengguna dalam menyiapkan laporan data pencatatan obat berdasarkan data obat keluar yang disimpan melalui sistem informasi.
SIKP-N-11	Informasi di dalam sistem sebaiknya dapat diakses sesuai dengan identitas pengguna sistem.	Kepala poliklinik, dokter, perawat, petugas farmasi, petugas administrasi.	Tidak ada.	Sistem informasi menyediakan layanan untuk membatasi akses terhadap informasi tertentu sesuai dengan hak akses yang dimiliki oleh pengguna sistem informasi.
SIKP-N-12	Sistem sebaiknya menyediakan layanan untuk pengelolaan data inisial yang harus dimuat sebelum sistem dapat	Peneliti.	Tidak ada,	Sistem informasi menyediakan layanan untuk pemuatan data inisial, diantaranya data wilayah

Tabel 4.4 Hasil Identifikasi Kebutuhan Pengguna

Kode Kebutuhan	Kebutuhan Pengguna	Pemangku Kepentingan	Situasi Saat Ini	Solusi Yang Ditawarkan
	digunakan pertama kali.			administratif Indonesia.

4.2.4 Identifikasi Pengguna

Identifikasi pengguna merupakan tahap yang dilakukan untuk mengetahui individu yang akan menggunakan sistem informasi secara langsung. Pengguna sistem merupakan bagian dari kelompok kategori pengguna dalam tipe pemangku kepentingan.

Hasil identifikasi pengguna diperoleh berdasarkan contoh pemangku kepentingan yang tergabung dalam kategori pengguna dalam tipe pemangku kepentingan. Penjelasan hasil identifikasi pengguna terdapat pada Tabel 4.5. Informasi mengenai identifikasi pengguna digunakan untuk melakukan identifikasi aktor yang akan dimodelkan ke dalam *use case diagram*.

Tabel 4.5 Hasil Identifikasi Pengguna

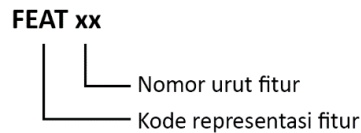
Tipe Stakeholder	Tipe Pengguna	Deskripsi
Pengguna	Kepala Poliklinik	Petugas Poliklinik Pabrik Gula Kebon Agung yang berprofesi sebagai dokter dan memiliki kedudukan tertinggi pada struktur organisasi poliklinik.
	Dokter	Petugas Poliklinik Pabrik Gula Kebon Agung yang berprofesi sebagai dokter, yang bertugas untuk memberikan diagnosis dan intervensi kesehatan kepada pasien.
	Perawat	Petugas Poliklinik Pabrik Gula Kebon Agung yang berprofesi sebagai perawat dan bidan. Perawat bertugas untuk melakukan pemeriksaan anamnesis kepada pasien.
	Petugas Farmasi	Petugas Poliklinik Pabrik Gula Kebon Agung yang bertugas mengelola ketersediaan obat dan memberikan obat kepada pasien sesuai dengan resep obat.
	Petugas Administrasi	Petugas poliklinik yang bertugas untuk mengelola pendaftaran pasien, mengelola arsip data pengobatan, mencetak surat rujukan

Tabel 4.5 Hasil Identifikasi Pengguna

Tipe Stakeholder	Tipe Pengguna	Deskripsi
		berdasarkan arahan dokter, dan mengelola administrasi kegiatan cek darah pasien.

4.2.5 Identifikasi Fitur

Fitur yang teridentifikasi dalam penelitian ini merepresentasikan solusi-solusi yang ditawarkan oleh untuk memenuhi kebutuhan pengguna, sehingga masalah yang dihadapi oleh pemangku kepentingan dapat diselesaikan. Identifikasi fitur dilakukan berdasarkan hasil identifikasi pengguna. Masing-masing fitur pada penelitian ini memiliki kode sebagai identitas. Keterangan kodifikasi fitur terdapat dalam Gambar 4.7.

**Gambar 4.7 Kodifikasi Fitur**

Hasil identifikasi fitur akan digunakan sebagai informasi untuk melakukan identifikasi persyaratan fungsional dan persyaratan non-fungsional sistem informasi yang akan dibangun pada penelitian ini. Informasi mengenai identitas fitur dan penjelasan fitur terdapat pada Tabel 4.6.

Tabel 4.6 Hasil Identifikasi Fitur

Kode Fitur	Deskripsi
FEAT1	Sistem memungkinkan untuk mengenali identitas pengguna sistem dan membatasi akses pengguna terhadap informasi dan layanan-layanan yang disediakan oleh sistem berdasarkan hak akses pengguna.
FEAT2	Sistem memungkinkan digunakan untuk mengelola data dasar kesehatan keluarga yang terdiri dari layanan penambahan dan penyajian data.
FEAT3	Sistem memungkinkan digunakan untuk mengelola data pengobatan secara holistik yang terdiri dari layanan penambahan dan penyajian data. Pengelolaan data secara holistik menggantikan fungsi Kartu Berobat yang digunakan untuk menyimpan informasi anamnesis, hasil diagnosa, dan intervensi kesehatan yang diberikan kepada pasien. Layanan pengelolaan data pengobatan secara holistik juga memungkinkan untuk penyimpanan informasi faktor risiko penyakit dan faktor pemicu penyakit.

Tabel 4.6 Hasil Identifikasi Fitur

Kode Fitur	Deskripsi
	Informasi intervensi kesehatan yang dapat dikelola melalui fitur ini ialah informasi tindakan medis, rujukan, dan surat pengantar cek darah yang diberikan kepada pasien.
FEAT4	Sistem memungkinkan digunakan untuk memperoleh hasil penghitungan tingkat risiko penyakit dan tingkat stres berdasarkan data dasar kesehatan keluarga yang tersimpan di dalam sistem
FEAT5	Sistem memungkinkan digunakan untuk menampilkan isi modul penyakit, modul faktor risiko, dan modul faktor pemicu penyakit melalui aplikasi peramban.
FEAT6	Sistem memungkinkan digunakan untuk mengelola data persediaan obat yang terdiri dari layanan pencatatan obat masuk dan keluar, layanan pelaporan persediaan obat. Sistem juga memungkinkan untuk pencetakan laporan persediaan obat dalam format dokumen yang dapat dikelola kembali menggunakan aplikasi Microsoft Excel.
FEAT7	Sistem memungkinkan digunakan untuk mengelola data wilayah administratif Indonesia yang berisi daftar kelurahan, kecamatan, kabupaten, dan provinsi di Indonesia.

Masing-masing fitur pada penelitian akan dikelompokkan berdasarkan keterkaitan terhadap kebutuhan pengguna yang harus dipenuhi. Pengelompokan ini menggambarkan bahwa sebuah fitur merupakan solusi untuk memenuhi kebutuhan tertentu. Pengelompokan fitur dengan kebutuhan pengguna terdapat pada Tabel 4.7.

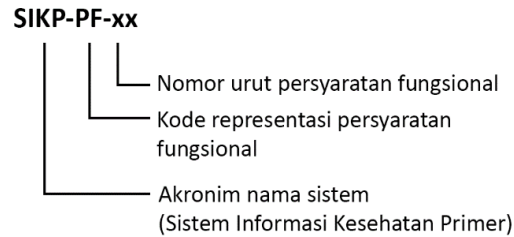
Tabel 4.7 Hubungan Kebutuhan Pengguna dengan Fitur

Kebutuhan Pengguna	Fitur
SIKP-N-11	FEAT1
SIKP-N-1	FEAT2
SIKP-N-2	FEAT3
SIKP-N-4	
SIKP-N-5	
SIKP-N-7	
SIKP-N-8	
SIKP-N-3	FEAT4
SIKP-N-6	FEAT5
SIKP-N-9	FEAT6

Tabel 4.7 Hubungan Kebutuhan Pengguna dengan Fitur

Kebutuhan Pengguna	Fitur
SIKP-N-10	
SIKP-N-12	FEAT7

4.2.6 Persyaratan Fungsional



Gambar 4.8 Kodifikasi Persyaratan Fungsional

Bagian ini menjelaskan persyaratan fungsional sistem yang akan dikembangkan dalam penelitian ini. Persyaratan fungsional merupakan kondisi atau kemampuan yang harus dipenuhi oleh sistem agar fitur sistem dapat berjalan dan dapat memenuhi kebutuhan pengguna. Tabel 4.8 menunjukkan hubungan antara fitur dengan persyaratan fungsional pada sistem informasi kesehatan yang akan dikembangkan. Persyaratan fungsional memiliki kode sesuai dengan kodifikasi dalam Gambar 4.8.

Tabel 4.8 Persyaratan Fungsional

Kode Fitur	Kode Persyaratan Fungsional	Deskripsi
FEAT1	SIKP-PF-01	Sistem dapat mengenali peran pengguna terhadap sistem berdasarkan username dan password yang dimasukkan oleh pengguna.
	SIKP-PF-02	Sistem dapat membatasi akses pengguna terhadap data dan fungsi sistem sesuai dengan peran pengguna.
FEAT2	SIKP-PF-03	Sistem dapat digunakan untuk melihat data dasar kesehatan keluarga yang tersimpan di sistem.
	SIKP-PF-04	Sistem dapat digunakan untuk menambah data dasar kesehatan keluarga.
	SIKP-PF-05	Sistem dapat digunakan untuk mengubah data dasar kesehatan keluarga yang tersimpan di sistem.
FEAT3	SIKP-PF-06	Sistem dapat digunakan untuk melihat data status pasien poliklinik yang tersimpan di sistem.
	SIKP-PF-07	Sistem dapat digunakan untuk menambah data status pasien poliklinik.
	SIKP-PF-08	Sistem dapat digunakan untuk mengubah data status pasien poliklinik yang tersimpan di sistem.
FEAT4	SIKP-PF-09	Sistem dapat digunakan untuk melihat data hasil diagnosis yang telah diberikan kepada pasien, yang tersimpan di sistem.

Tabel 4.8 Persyaratan Fungsional (lanjutan)

Kode Fitur	Kode Persyaratan Fungsional	Deskripsi
	SIKP-PF-10	Sistem dapat digunakan untuk menambah data hasil diagnosis yang telah diberikan kepada pasien.
	SIKP-PF-11	Sistem dapat digunakan untuk melihat data faktor risiko penyakit pasien yang tersimpan di sistem.
	SIKP-PF-12	Sistem dapat digunakan untuk menambah data faktor risiko penyakit pasien.
	SIKP-PF-13	Sistem dapat digunakan untuk melihat data faktor pemicu penyakit pasien yang tersimpan di sistem.
	SIKP-PF-14	Sistem dapat digunakan untuk menambah data faktor pemicu penyakit pasien.
	SIKP-PF-15	Sistem dapat digunakan untuk melihat data tindakan medis yang pernah diberikan kepada pasien.
	SIKP-PF-16	Sistem dapat digunakan untuk menambah data tindakan medis yang diberikan kepada pasien.
	SIKP-PF-17	Sistem dapat digunakan untuk melihat data rujukan yang pernah diberikan kepada pasien.
	SIKP-PF-18	Sistem dapat digunakan untuk menambah data rujukan yang diberikan kepada pasien.
	SIKP-PF-19	Sistem dapat digunakan untuk melihat data cek darah yang pernah diberikan kepada pasien.
	SIKP-PF-20	Sistem dapat digunakan untuk menambah data pengantar cek darah kepada pasien.
FEAT5	SIKP-PF-21	Sistem dapat digunakan untuk melihat isi modul penyakit yang berisi panduan cara mencegah dan mengatasi penyakit berdasarkan penyakit yang diderita pasien.
FEAT6	SIKP-PF-22	Sistem dapat digunakan untuk melihat isi modul faktor risiko penyakit yang berisi panduan cara mencegah dan mengatasi penyakit berdasarkan faktor risiko penyakit pasien.
FEAT7	SIKP-PF-23	Sistem dapat digunakan untuk melihat isi modul faktor pemicu penyakit yang berisi panduan cara mencegah dan mengatasi penyakit berdasarkan faktor pemicu penyakit pasien.
FEAT8	SIKP-PF-24	Sistem dapat digunakan untuk melihat data pengantar cek darah pasien.
	SIKP-PF-25	Sistem dapat digunakan untuk menambah data pengantar cek darah pasien.
FEAT9	SIKP-PF-26	Sistem dapat digunakan untuk merekam data obat keluar.
	SIKP-PF-27	Sistem dapat digunakan untuk memeriksa ketersediaan obat dan status pembiayaan obat.
FEAT10	SIKP-PF-28	Sistem dapat digunakan untuk menghitung tingkat risiko penyakit pasien berdasarkan data dasar kesehatan keluarga.
	SIKP-PF-29	Sistem dapat digunakan untuk menghitung tingkat stres pasien berdasarkan data dasar kesehatan keluarga.
	SIKP-PF-30	Sistem dapat menampilkan hasil perhitungan tingkat risiko penyakit pasien.
	SIKP-PF-31	Sistem dapat menampilkan hasil perhitungan tingkat stres pasien.

4.3 Pemodelan Use Case

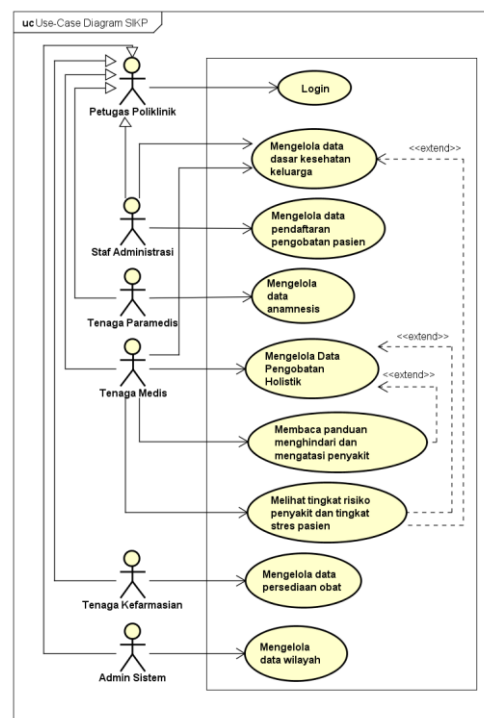
Pemodelan *use case* dilakukan setelah tahap analisis persyaratan. Pada pemodelan *use case*, dilakukan identifikasi aktor, identifikasi *use case*, dan spesifikasi untuk masing-masing *use case*. Identifikasi aktor dilakukan dengan menentukan pengelompokan pengguna sistem berdasarkan karakteristik yang dimiliki.

Selanjutnya, identifikasi *use case* dilakukan dengan menentukan tujuan aktor ketika menggunakan sistem. Penentuan tujuan aktor dilakukan dengan melakukan analisis terhadap hasil identifikasi kebutuhan pengguna yang telah dilakukan pada tahap analisis persyaratan.

Masing-masing *use case* yang teridentifikasi diberi penjelasan singkat, kondisi yang dibutuhkan sebelum sebuah *use case* dilakukan dan bagaimana kondisi setelah *use case* dilakukan, serta urutan tahap yang harus dilalui untuk menyelesaikan *use case*. Informasi tersebut didokumentasikan ke dalam spesifikasi *use case*. Alur tahap *use case* yang ada pada spesifikasi *use case* akan menjadi informasi untuk pemodelan aktivitas ke dalam *activity diagram*.

4.3.1 Use Case Diagram

Use case diagram menggambarkan hubungan aktor dengan *use case*, masing-masing aktor dan *use case* yang teridentifikasi berdasarkan hasil analisis persyaratan yang telah dilakukan pada penelitian ini. *Use case diagram* terdapat dalam Gambar 4.9.



Gambar 4.9 Use-Case Diagram Sistem Informasi Kesehatan “ANDAL”

Use case yang teridentifikasi, selanjutnya dihubungkan dengan pemangku kepentingan yang telah teridentifikasi sebelumnya. Hal ini dilakukan untuk membantu pemangku kepentingan memahami apa saja yang dapat dilakukan pengguna melalui sistem informasi yang akan dibangun. Hubungan *use case* dengan pemangku kepentingan terdapat pada Tabel 4.9.

Tabel 4.9 Hubungan *Use Case* dengan Pemangku Kepentingan

<i>Use Case</i>	Pengguna	Tipe Pemangku Kepentingan
Login.	Kepala Poliklinik, Dokter, Perawat, Petugas Farmasi, Petugas Administrasi.	Pengguna.
Mengelola data dasar kesehatan keluarga.	Petugas Administrasi, Dokter, Kepala Poliklinik.	
Mengelola data pendaftaran pengobatan pasien.	Petugas Administrasi.	
Mengelola data anamnesis.	Perawat	
Mengelola data pengobatan holistik.	Kepala Poliklinik, Dokter.	
Membaca panduan menghindari dan mengatasi penyakit.	Kepala Poliklinik, Dokter	
Melihat tingkat risiko penyakit dan risiko pemicu penyakit pasien.	Kepala Poliklinik, Dokter.	
Mengelola data persediaan obat.	Petugas Farmasi.	
Mengelola data wilayah.	Peneliti.	Pengembang.

Selain itu, masing-masing *use case* yang teridentifikasi juga dihubungkan dengan fitur yang teridentifikasi pada penelitian ini. Hal ini dilakukan untuk mempertegas bahwa hasil pemodelan *use case* dilakukan sesuai dengan hasil analisis persyaratan dalam penelitian ini. Hubungan fitur dengan *use case* terdapat pada Tabel 4.10.

Tabel 4.10 Hubungan *Use Case* dengan Fitur

<i>Use Case</i>	Fitur
Login	FEAT1

Tabel 4.10 Hubungan *Use Case* dengan Fitur

<i>Use Case</i>	Fitur
Mengelola data dasar kesehatan keluarga	FEAT2
Mengelola data pendaftaran pengobatan pasien.	FEAT3
Mengelola data anamnesis.	
Mengelola data pengobatan holistik.	
Membaca panduan menghindari dan mengatasi penyakit.	FEAT5
Melihat tingkat risiko penyakit dan risiko pemicu penyakit pasien.	FEAT4
Mengelola data persediaan obat.	FEAT6
Mengelola data wilayah.	FEAT7

4.3.2 Deskripsi Aktor

Deskripsi aktor merupakan penjelasan singkat mengenai tanggung jawab aktor yang telah teridentifikasi di dalam penelitian ini. Di dalam deskripsi aktor juga terdapat informasi singkat mengenai tujuan aktor dalam menggunakan sistem informasi yang akan dibangun dalam penelitian ini. Deskripsi aktor Petugas Poliklinik, Petugas Administrasi, Tenaga Paramedis, Tenaga Medis, Tenaga Kefarmasian, Admin Sistem terdapat pada Tabel 4.11.

Tabel 4.11 Deskripsi Aktor

Nama Aktor	Deskripsi
Petugas Poliklinik	Aktor Petugas Poliklinik diperankan oleh seluruh petugas Poliklinik Pabrik Gula Kebon Agung yang akan menggunakan sistem secara langsung. Petugas Poliklinik dapat menggunakan sistem dan mengakses informasi sesuai dengan hak akses yang dikenali oleh sistem.
Staf Administrasi	Aktor Staf Administrasi diperankan oleh petugas administrasi. Aktor ini dapat mengelola data dasar kesehatan keluarga dan mengelola data pendaftaran pengobatan pasien melalui sistem.
Tenaga Paramedis	Aktor Tenaga Paramedis diperankan oleh perawat dan bidan Poliklinik Pabrik Gula Kebon Agung. Aktor ini dapat mengelola data anamnesis melalui sistem.
Tenaga Medis	Aktor Tenaga Medis diperankan oleh dokter dan Kepala Poliklinik Pabrik Gula Kebon Agung. Aktor ini dapat mengelola data pengobatan holistik, membaca modul pencegahan dan

Tabel 4.11 Deskripsi Aktor

Nama Aktor	Deskripsi
	penanganan penyakit, serta melihat tingkat risiko penyakit dan tingkat stres pasien melalui sistem.
Tenaga Kefarmasian	Aktor Tenaga Kefarmasian diperankan oleh petugas farmasi Poliklinik Pabrik Gula Kebon Agung. Aktor ini dapat mengelola data persediaan obat melalui sistem.
Admin Sistem	Aktor Admin diperankan oleh peneliti. Aktor ini dapat mengelola data wilayah administratif melalui sistem.

4.3.3 Spesifikasi Use Case

Spesifikasi *use case* berisi penjelasan singkat mengenai setiap *use case* yang teridentifikasi dalam penelitian ini. Spesifikasi juga memuat informasi mengenai aktor yang menjalankan *use case*, kondisi sebelum dan sesudah *use case* dijalankan, serta tahap-tahap yang dilakukan ketika *use case* dilakukan. Tahap-tahap pada spesifikasi *use case* digunakan sebagai panduan memodelkan aktivitas ke dalam *activity diagram*.

4.3.3.1 Spesifikasi Use Case Login

Tabel 4.12 Spesifikasi Use Case Login

Brief Description	<i>Use case</i> Login menjelaskan bagaimana aktor Petugas Poliklinik menggunakan sistem untuk melakukan verifikasi terhadap identitasnya. Sehingga Petugas Poliklinik dapat mengakses informasi pada sistem sesuai dengan hak aksesnya.
Actor	Petugas Poliklinik
Pre-condition	<ul style="list-style-type: none"> Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. Sistem terhubung dengan <i>server</i>.
Post-condition	<ul style="list-style-type: none"> Identitas aktor Petugas Poliklinik teridentifikasi oleh sistem. Sistem menampilkan informasi sesuai hak akses aktor Petugas Poliklinik.
Basic Flow	<p>{use case dimulai}</p> <ol style="list-style-type: none"> Aktor Petugas Poliklinik memasukkan <i>username</i> dan <i>password</i>. Aktor Petugas Poliklinik memilih untuk masuk ke dalam sistem. Sistem mengidentifikasi identitas pengguna. <p>{mengidentifikasi pengguna}</p>

Tabel 4.12 Spesifikasi Use Case Login

	<p>4. Sistem menyimpan sesi pengguna ke dalam sistem. {menyimpan sesi pengguna}</p> <p>5. Sistem menampilkan informasi berdasarkan hak akses Aktor Petugas Poliklinik. {use case selesai}</p> <p>6. <i>Use case</i> selesai.</p>
Alternative Flow	<p>A1. Menangani Kesalahan Pengisian Username dan Password.</p> <p>Pada {mengidentifikasi pengguna}, jika sistem mendeteksi kesalahan pengisian <i>username</i> dan <i>password</i>, maka sistem menampilkan pesan kesalahan dalam pengisian <i>username</i> dan <i>password</i>, kemudian <i>use case</i> dilanjutkan pada {use case dimulai}.</p> <p>A2. Menangani Kegagalan Identifikasi Pengguna.</p> <p>Pada {mengidentifikasi pengguna}, jika sistem mengalami kegagalan identifikasi pengguna, maka sistem menampilkan pesan bahwa identifikasi pengguna gagal, kemudian <i>use case</i> selesai.</p> <p>A3. Menangani Kegagalan Penyimpanan Sesi Pengguna.</p> <p>Pada {menyimpan sesi pengguna}, jika sistem mengalami kegagalan penyimpanan sesi pengguna, maka sistem menampilkan pesan bahwa sesi pengguna gagal disimpan, kemudian <i>use case</i> selesai.</p>

4.3.3.2 Spesifikasi Use Case Mengelola Data Dasar Kesehatan Keluarga

Tabel 4.13 Spesifikasi Use Case Mengelola Data Dasar Kesehatan Keluarga

Brief Description	<p><i>Use Case</i> Mengelola Data Dasar Kesehatan Keluarga menjelaskan bagaimana aktor Staf Administrasi menggunakan sistem untuk menambah data dasar kesehatan keluarga, mengubah sebagian data dasar kesehatan keluarga, menghapus data dasar kesehatan keluarga, dan melihat data dasar kesehatan keluarga yang tersimpan di dalam sistem.</p>
Actor	<p>Staf Administrasi, Tenaga Medis</p>
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Staf Administrasi atau Tenaga Medis berhasil.

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<ul style="list-style-type: none"> • Sesi <i>login</i> aktor Staf Administrasi atau Tenaga Medis tersimpan dalam sistem. • Data wilayah telah disimpan di dalam sistem.
Post-condition	<ul style="list-style-type: none"> • Sistem berhasil menyimpan data dasar kesehatan keluarga. • Aktor Staf Administrasi atau Tenaga Medis dapat melihat data dasar kesehatan keluarga yang telah berhasil disimpan.
Extension	<i>Extensions use case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien pada {melihat detail data dasar} di basic flow, pada {menampilkan detail data dasar} di alternative flow.</i>
Basic Flow	<p>{use case dimulai}</p> <p>{melihat data dasar}</p> <ol style="list-style-type: none"> 1. Lakukan <i>subflow</i> Menambah Data Identitas Pasien. 2. Lakukan <i>subflow</i> Menambah Data Perkawinan. 3. Jalankan <i>subflow</i> Menambah Data Anggota Keluarga. 4. Jalankan <i>subflow</i> Menambah Data Ekonomi. 5. Jalankan <i>subflow</i> Menambah Data Perilaku Kesehatan dan Keselamatan. 6. Jalankan <i>subflow</i> Menambah Data Riwayat Kesehatan. 7. Jalankan <i>subflow</i> Menambah Data Kuesioner Perhitungan Tingkat Stres. <p>{menampilkan data dasar}</p> <ol style="list-style-type: none"> 8. Sistem menampilkan data dasar kesehatan keluarga yang disimpan di dalam sistem. <p>{melihat detail data dasar}</p> <p>{use case selesai}</p> <ol style="list-style-type: none"> 9. <i>Use case</i> selesai.
Alternative Flow	<p>A1. Membatalkan Pengisian Formulir.</p> <p>Pada {membatalkan penyimpanan data identitas pasien}, {membatalkan penyimpanan data perkawinan}, {membatalkan penyimpanan data anggota keluarga}, {membatalkan penyimpanan data ekonomi}, {membatalkan penyimpanan data perilaku kesehatan dan keselamatan}, {membatalkan penyimpanan data riwayat kesehatan}, {membatalkan penyimpanan data kuesioner} pada <i>subflow</i>,</p>

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<p>jika Aktor Staf Administrasi memilih untuk membatalkan pengisian formulir, maka:</p> <ol style="list-style-type: none"> 1. Sistem akan mengosongkan formulir yang telah terisi. 2. <i>Use case</i> dilanjutkan pada. <p>A2. Menampilkan Data Dasar.</p> <p>Pada {melihat data dasar} pada <i>basic flow</i>, jika aktor Staf Administrasi atau Tenaga Medis ingin melihat data dasar kesehatan yang tersimpan di dalam sistem, maka <i>use case</i> dilanjutkan pada {menampilkan data dasar}.</p> <p>A3. Menampilkan Detail Data Dasar.</p> <p>Pada {melihat detail data dasar} pada <i>basic flow</i>, jika aktor Staf Administrasi atau Tenaga Medis ingin melihat detail data dasar kesehatan, maka:</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi atau Tenaga Medis memilih data dasar kesehatan keluarga yang ingin ditampilkan detailnya. 2. Sistem melakukan proses pemuatan data. {memuat detail data dasar} 3. Sistem menampilkan detail data dasar kesehatan keluarga. {menampilkan detail data dasar} {mengubah data dasar} 4. <i>Use case</i> selesai. <p>A4. Mengubah Data Identitas Pasien.</p> <p>Pada {mengubah data dasar} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin mengubah data identitas pasien, maka lakukan <i>subflow</i> Mengubah Data Identitas Pasien.</p> <p>A5. Mengubah Data Perkawinan.</p> <p>Pada {mengubah data dasar} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin mengubah data perkawinan, maka lakukan <i>subflow</i> Mengubah Data Perkawinan.</p> <p>A6. Mengubah Data Anggota Keluarga.</p> <p>Pada {mengubah data dasar} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin mengubah data anggota keluarga, maka lakukan <i>subflow</i> Mengubah Data Anggota Keluarga.</p>
--	---

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<p>A7. Mengubah Data Ekonomi.</p> <p>Pada {mengubah data dasar} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin mengubah data ekonomi, maka lakukan <i>subflow</i> Mengubah Data Ekonomi.</p> <p>A8. Menghapus Data Dasar Kesehatan Keluarga.</p> <p>Pada {melihat detail data dasar}, jika aktor Staf Administrasi atau Tenaga Medis ingin menghapus data dasar kesehatan keluarga, maka:</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi atau Tenaga Medis memilih untuk menghapus salah satu data dasar kesehatan keluarga. 2. Sistem melakukan proses penghapusan data. {menghapus data dasar} 3. Sistem menampilkan pesan bahwa proses penghapusan pesan berhasil. 4. <i>Use case</i> selesai. <p>A9. Menghapus Data Perilaku Kesehatan dan Keselamatan.</p> <p>Pada {melihat detail data dasar} pada <i>basic flow</i>, jika aktor Staf Administrasi dan Tenaga Medis ingin menghapus data perilaku kesehatan dan keselamatan, maka lakukan <i>subflow</i> Menghapus Data Perilaku Kesehatan dan Keselamatan.</p> <p>A10. Menghapus Data Riwayat Kesehatan.</p> <p>Pada {melihat detail data dasar} pada <i>basic flow</i>, jika aktor Staf Administrasi dan Tenaga Medis ingin menghapus data riwayat kesehatan, maka lakukan <i>subflow</i> Menghapus Data Riwayat Kesehatan.</p> <p>A11. Menghapus Data Kuesioner Perhitungan Tingkat Stres.</p> <p>Pada {melihat detail data dasar} pada <i>basic flow</i>, jika aktor Staf Administrasi dan Tenaga Medis ingin menghapus data kuesioner perhitungan tingkat stres, maka lakukan <i>subflow</i> Menghapus Data Kuesioner Perhitungan Tingkat Stres.</p> <p>A12. Menangani Kondisi Ketika Data Tidak Tersedia.</p> <p>Pada {menampilkan data dasar} pada <i>basic flow</i>, jika data dasar kesehatan keluarga belum tersedia, maka sistem menampilkan pesan bahwa data belum tersedia, kemudian <i>use case</i> selesai.</p> <p>A13. Menangani Kegagalan Penyimpanan Data Baru.</p>
--	--

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<p>Pada {menyimpan data identitas pasien}, {menyimpan data perkawinan}, {menyimpan data anggota keluarga}, {menyimpan data ekonomi}, {menyimpan data perilaku kesehatan dan keselamatan}, {menyimpan data riwayat kesehatan}, {menyimpan data kuesioner} pada <i>subflow</i>, jika sistem mengalami kegagalan penyimpanan, maka sistem menampilkan pesan kegagalan, kemudian <i>use case</i> selesai.</p> <p>A14. Menangani Kegagalan Sistem Ketika Memuat Data Yang Akan Diubah.</p> <p>Pada {memuat data pasien}, {memuat data perkawinan}, {memuat data anggota keluarga}, {memuat data ekonomi} pada <i>subflow</i>, jika sistem mengalami kegagalan pemuatan data yang akan diubah oleh aktor Staf Administrasi atau Tenaga Medis, maka sistem akan menampilkan pesan bahwa data gagal dimuat, kemudian <i>use case</i> selesai.</p> <p>A15. Menangani Kegagalan Pengubahan Data.</p> <p>Pada {memproses perubahan data identitas pasien}, {memproses perubahan data perkawinan}, {memproses perubahan data anggota keluarga}, {memproses perubahan data ekonomi} pada <i>subflow</i>, jika sistem mengalami kegagalan ketika memproses perubahan data, maka sistem akan menampilkan pesan kegagalan, kemudian <i>use case</i> selesai.</p> <p>A16. Menangani Kegagalan Penghapusan Data Dasar.</p> <p>Pada {menghapus data dasar} pada <i>alternative flow</i> atau pada {menghapus data perilaku kesehatan dan keselamatan}, {menghapus data riwayat kesehatan}, {menghapus data kuesioner perhitungan tingkat stres} pada <i>subflow</i>, jika sistem mengalami kegagalan dalam proses penghapusan data dasar kesehatan keluarga, maka sistem akan menampilkan pesan kegagalan penghapusan data, kemudian <i>use case</i> selesai.</p>
Subflow	<p>S1. Menambah Data Identitas Pasien.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data identitas pasien. 2. Aktor Staf Administrasi memilih untuk menyimpan data identitas pasien. {membatalkan penyimpanan data identitas pasien}

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<ol style="list-style-type: none"> 3. Sistem melakukan proses penyimpanan data identitas pasien. {menyimpan data identitas pasien} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. Use case dilanjutkan ke langkah berikutnya. <p>S2. Menambah Data Perkawinan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data perkawinan. 2. Aktor Staf Administrasi memilih untuk menyimpan data perkawinan. {membatalkan penyimpanan data perkawinan} 3. Sistem melakukan proses penyimpanan data perkawinan. {menyimpan data perkawinan} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. Use case dilanjutkan ke langkah berikutnya. <p>S3. Menambah Data Anggota Keluarga.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data anggota keluarga. 2. Aktor Staf Administrasi memilih untuk menyimpan data anggota keluarga. {membatalkan penyimpanan data anggota keluarga} 3. Sistem melakukan proses penyimpanan data anggota keluarga. {menyimpan data anggota keluarga} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. Use case dilanjutkan ke langkah berikutnya. <p>S4. Menambah Data Ekonomi.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data ekonomi. 2. Aktor Staf Administrasi memilih untuk menyimpan data ekonomi. {membatalkan penyimpanan data ekonomi} 3. Sistem melakukan proses penyimpanan data ekonomi. {menyimpan data ekonomi} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. Use case dilanjutkan ke langkah berikutnya.
--	---

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<p>S5. Menambah Data Perilaku Kesehatan dan Keselamatan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data perilaku kesehatan dan keselamatan. 2. Aktor Staf Administrasi memilih untuk menyimpan data perilaku kesehatan dan keselamatan. {membatalkan penyimpanan data perilaku kesehatan dan keselamatan} 3. Sistem melakukan proses penyimpanan data perilaku kesehatan dan keselamatan. {menyimpan data perilaku kesehatan dan keselamatan} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. <i>Use case</i> dilanjutkan ke langkah berikutnya. <p>S6. Menambah Data Riwayat Kesehatan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data riwayat kesehatan. 2. Aktor Staf Administrasi memilih untuk menyimpan data riwayat kesehatan. {membatalkan penyimpanan data riwayat kesehatan} 3. Sistem melakukan proses penyimpanan data riwayat kesehatan. {menyimpan data riwayat kesehatan} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. <i>Use case</i> dilanjutkan ke langkah berikutnya. <p>S7. Menambah Data Kuesioner Perhitungan Tingkat Stres.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memasukkan data kuesioner. 2. Aktor Staf Administrasi memilih untuk menyimpan data kuesioner. {membatalkan penyimpanan data kuesioner} 3. Sistem melakukan proses penyimpanan data kuesioner. {menyimpan data kuesioner} 4. Sistem menampilkan pesan penyimpanan data berhasil. 5. <i>Use case</i> dilanjutkan ke langkah berikutnya. <p>S8. Menambah Data Identitas Pasien.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih untuk mengubah data identitas pasien.
--	--

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<ol style="list-style-type: none"> 2. Sistem memuat data identitas pasien yang akan diubah. {memuat data pasien} 3. Sistem menampilkan data identitas pasien yang akan diubah. 4. Aktor Staf Administrasi mengubah data identitas pasien. 5. Aktor Staf Administrasi memilih untuk menyimpan perubahan data. 6. Sistem memproses perubahan data. {memproses perubahan data identitas pasien} 7. Sistem menampilkan pesan bahwa data identitas pasien berhasil diubah. 8. <i>Use case</i> selesai. <p>S9. Mengubah Data Perkawinan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih untuk mengubah data perkawinan. 2. Sistem memuat data perkawinan yang akan diubah. {memuat data perkawinan} 3. Sistem menampilkan data perkawinan yang akan diubah. 4. Aktor Staf Administrasi mengubah data perkawinan. 5. Aktor Staf Administrasi memilih untuk menyimpan perubahan data. 6. Sistem memproses perubahan data. {memproses perubahan data perkawinan} 7. Sistem menampilkan pesan bahwa data perkawinan berhasil diubah. 8. <i>Use case</i> selesai. <p>S10. Mengubah Data Anggota Keluarga.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih untuk mengubah data anggota keluarga. 2. Sistem memuat data anggota keluarga yang akan diubah. {memuat data anggota keluarga} 3. Sistem menampilkan data anggota keluarga yang akan diubah. 4. Aktor Staf Administrasi mengubah data anggota keluarga.
--	--

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Dasar Kesehatan Keluarga

	<ol style="list-style-type: none"> 5. Aktor Staf Administrasi memilih untuk menyimpan perubahan data. 6. Sistem memproses perubahan data. {memproses perubahan data anggota keluarga} 7. Sistem menampilkan pesan bahwa data anggota keluarga berhasil diubah. 8. <i>Use case</i> selesai. <p>S11. Mengubah Data Ekonomi.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih untuk mengubah data ekonomi. 2. Sistem memuat data ekonomi yang akan diubah. {memuat data ekonomi} 3. Sistem menampilkan data ekonomi yang akan diubah. 4. Aktor Staf Administrasi mengubah data ekonomi. 5. Aktor Staf Administrasi memilih untuk menyimpan perubahan data. 6. Sistem memproses perubahan data. {memproses perubahan data ekonomi} 7. Sistem menampilkan pesan bahwa data ekonomi berhasil diubah. 8. <i>Use case</i> selesai. <p>S12. Menghapus Data Perilaku Kesehatan dan Keselamatan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi atau Tenaga Medis memilih untuk menghapus data perilaku kesehatan dan keselamatan. 2. Sistem memproses penghapusan data perilaku kesehatan dan keselamatan. {menghapus data perilaku kesehatan dan keselamatan} 3. Sistem menampilkan pesan bahwa data perilaku kesehatan dan keselamatan berhasil dihapus. 4. <i>Use case</i> selesai. <p>S13. Menghapus Data Riwayat Kesehatan.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi atau Tenaga Medis memilih untuk menghapus data riwayat kesehatan.
--	--

Tabel 4.13 Spesifikasi Use Case Mengelola Data Dasar Kesehatan Keluarga

	<ol style="list-style-type: none"> 2. Sistem memproses penghapusan data riwayat kesehatan. {menghapus data riwayat kesehatan} 3. Sistem menampilkan pesan bahwa data riwayat kesehatan berhasil dihapus. 4. <i>Use case</i> selesai. <p>S14. Menghapus Data Kuesioner Perhitungan Tingkat Stres.</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi atau Tenaga Medis memilih untuk menghapus data kuesioner perhitungan tingkat stres. 2. Sistem memproses penghapusan data kuesioner perhitungan tingkat stres. {menghapus data kuesioner perhitungan tingkat stres} 3. Sistem menampilkan pesan bahwa data kuesioner perhitungan tingkat stres berhasil dihapus. 4. <i>Use case</i> selesai.
--	--

4.3.3.3 Spesifikasi Use Case Mengelola Data Pendaftaran Pengobatan Pasien

Tabel 4.14 Spesifikasi Use Case Mengelola Data Pendaftaran Pengobatan Pasien

Brief Description	Use case Mengelola Data Pendaftaran Pengobatan Pasien menunjukkan bagaimana aktor Staf Administrasi menggunakan sistem untuk mendaftarkan pasien yang datang untuk berobat dan melihat data pasien yang data berobat ke poliklinik.
Actor	Staf Administrasi.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Staf Administrasi berhasil. • Sesi <i>login</i> aktor Staf Administrasi tersimpan dalam sistem. • Data identitas pasien yang datang untuk melakukan pengobatan sudah tersimpan di dalam sistem. • Data dokter dan poli yang tersedia di poliklinik telah tersimpan di dalam sistem.
Post-condition	<ul style="list-style-type: none"> • Sistem berhasil menyimpan data pendaftaran pengobatan pasien.

Tabel 4.14 Spesifikasi *Use Case* Mengelola Data Pendaftaran Pengobatan Pasien

	<ul style="list-style-type: none"> Aktor Staf Administrasi dapat melihat data pendaftaran pengobatan pasien yang telah berhasil disimpan.
Basic Flow	<p>{use case dimulai}</p> <p>{melihat data pendaftaran pengobatan}</p> <ol style="list-style-type: none"> Sistem menampilkan data identitas pasien yang tersimpan di dalam sistem. Sistem menampilkan data poli yang tersimpan di dalam sistem. Sistem menampilkan data iden Aktor Staf Administrasi memilih satu data pasien. Aktor Staf Adminstrasi memilih poli untuk pasien. Aktor Staf Administrasi memilih untuk menyimpan data pendaftaran pengobatan pasien. Sistem melakukan proses untuk menyimpan data. <p>{menyimpan data pendaftaran}</p> <ol style="list-style-type: none"> Sistem menampilkan pesan bahwa pendaftaran pasien berhasil dilakukan. Sistem menampilkan nomor pendaftaran pasien. <p>{menampilkan nomor pendaftaran}</p> <p>{use case selesai}</p> <ol style="list-style-type: none"> <i>Use case</i> selesai.
Alternative Flow	<p>A1. Melihat Data Pendaftaran Pengobatan Harian.</p> <p>Pada {melihat data pendaftaran pengobatan} pada <i>basic flow</i>, jika aktor Staf Administrasi ingin melihat data pendaftaran pengobatan harian, maka:</p> <ol style="list-style-type: none"> Aktor Staf Administrasi memilih untuk menampilkan data pendaftaran pengobatan harian. Sistem melakukan proses pemuatan data pendaftaran pengobatan harian. <p>{memuat data pendaftaran pengobatan harian}</p> <ol style="list-style-type: none"> Sistem menampilkan data pendaftaran pengobatan harian. <p>{menampilkan data pendaftaran pengobatan harian}</p> <ol style="list-style-type: none"> <i>Use case</i> selesai. <p>A2. Melihat Seluruh Data Pendaftaran Pengobatan.</p>

Tabel 4.14 Spesifikasi *Use Case* Mengelola Data Pendaftaran Pengobatan Pasien

	<p>Pada {melihat data pendaftaran pengobatan} pada <i>basic flow</i>, jika aktor Staf Administrasi ingin melihat seluruh data pendaftaran pengobatan yang tersimpan di dalam sistem, maka:</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih untuk menampilkan seluruh data pendaftaran pengobatan. 2. Sistem melakukan proses pemuatan data pendaftaran pengobatan. {memuat data pendaftaran pengobatan} 3. Sistem menampilkan seluruh data pendaftaran pengobatan. {menampilkan seluruh data pendaftaran pengobatan} 4. <i>Use case</i> selesai. <p>A3. Melihat Detail Data Pendaftaran Pengobatan.</p> <p>Pada {menampilkan data pendaftaran pengobatan harian}, {menampilkan seluruh data pendaftaran pengobatan} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin melihat detail data pendaftaran pengobatan, maka:</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih satu data pendaftaran pengobatan untuk ditampilkan detailnya. 2. Sistem melakukan proses pemuatan detail data pendaftaran pengobatan. {memuat detail data pendaftaran pengobatan} 3. Sistem menampilkan detail data pendaftaran pengobatan. 4. <i>Use case</i> selesai. <p>A4. Menghapus Data Pendaftaran Pengobatan Pasien.</p> <p>Pada {menampilkan data pendaftaran pengobatan harian}, {menampilkan seluruh data pendaftaran pengobatan} pada <i>alternative flow</i>, jika aktor Staf Administrasi ingin menghapus data pendaftaran pengobatan pasien, maka:</p> <ol style="list-style-type: none"> 1. Aktor Staf Administrasi memilih satu data pendaftaran pengobatan pasien untuk dihapus. 2. Sistem melakukan proses penghapusan data pendaftaran pengobatan pasien. {menghapus data pendaftaran pengobatan pasien}
--	--

Tabel 4.14 Spesifikasi *Use Case* Mengelola Data Pendaftaran Pengobatan Pasien

	<p>3. Sistem menampilkan pesan bahwa proses penghapusan data pendaftaran pengobatan pasien berhasil dihapus.</p> <p>4. <i>Use case</i> selesai.</p> <p>A5. Menangani Kegagalan Penyimpanan Data Pengobatan Pasien.</p> <p>Pada {menyimpan data pendaftaran} pada <i>basic flow</i>, jika sistem mengalami kegagalan proses penyimpanan data pendaftaran pengobatan pasien, maka sistem menampilkan pesan bahwa proses penyimpanan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A6. Menangani Kegagalan Penghapusan Data Pendaftaran Pengobatan Pasien.</p> <p>Pada {menghapus data pendaftaran pengobatan pasien} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penghapusan data, maka sistem menampilkan pesan bahwa penghapusan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A7. Menangani Kondisi Ketika Data Pendaftaran Pengobatan Tidak Tersedia.</p> <p>Pada {memuat data pendaftaran pengobatan harian}, {memuat data pendaftaran pengobatan} pada <i>alternative flow</i>, jika data pendaftaran pengobatan tidak tersedia, maka sistem menampilkan pesan bahwa data pengobatan pasien tidak tersedia, kemudian <i>use case</i> selesai.</p> <p>A8. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data pendaftaran pengobatan harian}, {memuat data pendaftaran pengobatan}, {memuat detail data pendaftaran pengobatan} pada <i>alternative flow</i>, jika sistem mengalami kegagalan pemuatan data pendaftaran pengobatan, maka sistem menampilkan pesan bahwa pemuatan data pendaftaran pengobatan gagal dilakukan, kemudian <i>use case</i> selesai.</p>
--	---

4.3.3.4 Spesifikasi *Use Case* Mengelola Data Anamnesis

Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Anamnesis

Brief Description	<p><i>Use Case</i> Mengelola Data Anamnesis menjelaskan bagaimana aktor Tenaga Paramedis menambahkan, mengubah, menghapus, dan melihat data tekanan darah (TD), frekuensi</p>
--------------------------	---

Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Anamnesis

	pernapasan (RR), denyut nadi (N), suhu badan, dan anamnesis pasien.
Actor	Tenaga Paramedis.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Tenaga Paramedis berhasil. • Sesi <i>login</i> aktor Tenaga Paramedis tersimpan dalam sistem. • Data pendaftaran pengobatan pasien telah tersimpan di dalam sistem.
Post-condition	<ul style="list-style-type: none"> • Sistem berhasil menyimpan data anamnesis pasien. • Aktor Tenaga Paramedis dapat melihat data anamnesis pasien yang telah berhasil disimpan.
Basic Flow	<p>{use case dimulai}</p> <p>{melihat seluruh data anamnesis}</p> <ol style="list-style-type: none"> 1. Sistem menampilkan data pendaftaran pengobatan pasien harian. {menampilkan data pendaftaran pengobatan pasien harian} 2. Aktor Tenaga Paramedis memilih satu data pendaftaran pengobatan pasien yang akan diberi anamnesis. 3. Sistem memuat data pendaftaran pengobatan pasien yang telah dipilih oleh aktor Tenaga Paramedis. {memuat data pendaftaran pengobatan terpilih} 4. Sistem menampilkan formulir untuk menambahkan data anamnesis. 5. Aktor Tenaga Paramedis mengisi data anamnesis. 6. Aktor Tenaga Paramedis memilih untuk menyimpan data anamnesis. 7. Sistem melakukan proses penyimpanan data anamnesis. {menyimpan data anamnesis} 8. Sistem menampilkan pesan bahwa penyimpanan data berhasil. {use case selesai} 9. <i>Use case</i> selesai.

Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Anamnesis

<p>Alternative Flow</p>	<p>A1. Melihat Data Anamnesis Harian.</p> <p>Jika aktor Tenaga Paramedis ingin melihat data anamnesis harian, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Paramedis memilih untuk melihat data anamnesis harian. 2. <i>Use case</i> dilanjutkan pada {melihat seluruh data anamnesis}. 3. <i>Use case</i> selesai pada {menampilkan data pendaftaran pengobatan pasien harian}. <p>A2. Melihat Seluruh Data Anamnesis.</p> <p>Pada {melihat seluruh data anamnesis} pada <i>basic flow</i>, jika aktor Tenaga Paramedis ingin melihat seluruh data anamnesis, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Paramedis memilih untuk melihat seluruh data anamnesis. 2. Sistem memuat seluruh data anamnesis. {memuat seluruh data anamnesis} 3. Sistem menampilkan seluruh data anamnesis. {menampilkan seluruh data anamnesis} 4. <i>Use case</i> selesai. <p>A3. Melihat Detail Data Anamnesis.</p> <p>Pada {menampilkan data pendaftaran pengobatan pasien harian} pada <i>basic flow</i>, pada {menampilkan seluruh data anamnesis} pada <i>alternative flow</i>, jika aktor Tenaga Paramedis ingin melihat detail data anamnesis, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Paramedis memilih satu data untuk dilihat detailnya. 2. Sistem memuat detail data anamnesis. {memuat detail data anamnesis} 3. Sistem menampilkan detail data anamnesis. {menampilkan detail data anamnesis} 4. <i>Use case</i> selesai. <p>A4. Mengubah Data Anamnesis.</p> <p>Pada {menampilkan data pendaftaran pengobatan pasien harian} pada <i>basic flow</i>, pada {menampilkan seluruh data</p>
--------------------------------	---

Tabel 4.15 Spesifikasi *Use Case* Mengelola Data Anamnesis

	<p>anamnesis} pada <i>alternative flow</i>, jika aktor Tenaga Paramedis ingin mengubah data anamnesis, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Paramedis memilih satu data anamnesis untuk diubah. 2. Sistem memuat data anamnesis yang akan diubah. {memuat data anamnesis yang akan diubah} 3. Sistem menampilkan data anamnesis yang akan diubah. 4. Aktor Tenaga Paramedis mengganti data anamnesis. 5. Aktor Tenaga Paramedis memilih untuk menyimpan perubahan data anamnesis. 6. Sistem melakukan proses pengubahan data anamnesis. {menyimpan perubahan data anamnesis} 7. Sistem menampilkan pesan bahwa pengubahan data berhasil. 8. <i>Use case</i> selesai. <p>A5. Menghapus Data Anamnesis Pasien.</p> <p>Pada {menampilkan data pendaftaran pengobatan pasien harian} pada <i>basic flow</i>, pada {menampilkan seluruh data anamnesis} pada <i>alternative flow</i>, jika aktor Tenaga Paramedis ingin menghapus data anamnesis, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Paramedis memilih satu data anamnesis untuk dihapus. 2. Sistem melakukan proses penghapusan data anamnesis. {menghapus data anamnesis} 3. Sistem menampilkan pesan bahwa proses penghapusan data anamnesis berhasil. 4. <i>Use case</i> selesai. <p>A6. Menangani Kegagalan Penambahan Data Anamnesis.</p> <p>Pada {menyimpan data anamnesis} pada <i>basic flow</i>, jika sistem mengalami kegagalan menyimpan data anamnesis baru, maka sistem menampilkan pesan bahwa proses penyimpanan data anamnesis gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A7. Menangani Kegagalan Pengubahan Data Anamnesis.</p> <p>Pada {menyimpan perubahan data anamnesis} pada <i>alternative flow</i>, jika sistem mengalami kegagalan menyimpan perubahan data anamnesis, maka sistem</p>
--	---

Tabel 4.15 Spesifikasi Use Case Mengelola Data Anamnesis

	<p>menampilkan pesan bahwa proses pengubahan data anamnesis gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A8. Menangani Kegagalan Penghapusan Data Anamnesis.</p> <p>Pada {menghapus data anamnesis} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penghapusan data anamnesis, maka sistem menampilkan pesan bahwa proses penghapusan data anamnesis gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A9. Menangani Kondisi Data Anamnesis Belum Tersedia.</p> <p>Pada {melihat seluruh data anamnesis} pada <i>basic flow</i>, pada {memuat seluruh data anamnesis} pada <i>alternative flow</i>, jika data anamnesis belum tersedia di dalam sistem, maka sistem menampilkan pesan bahwa data anamnesis belum tersedia, kemudian <i>use case</i> selesai.</p> <p>A10. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data pendaftaran pengobatan terpilih} pada <i>basic flow</i>, pada {memuat seluruh data anamnesis}, {memuat detail data anamnesis}, {memuat data anamnesis yang akan diubah} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam pemuatan data, maka sistem menampilkan pesan bahwa proses memuat data gagal dilakukan, kemudian <i>use case</i> selesai.</p>
--	---

4.3.3.5 Spesifikasi Use Case Mengelola Data Pengobatan Holistik

Tabel 4.16 Spesifikasi Use Case Mengelola Data Pengobatan Holistik

Brief Description	Use case Mengelola Data Pengobatan Holistik menjelaskan bagaimana aktor Tenaga Medis menambah, menghapus, dan melihat data diagnosis dan intervensi yang diberikan kepada pasien yang telah terdaftar untuk melakukan pengobatan di poliklinik.
Actor	Tenaga Medis.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan server. • Login yang dilakukan aktor Tenaga Medis berhasil. • Sesi login aktor Tenaga Medis tersimpan dalam sistem. • Data pendaftaran pengobatan pasien telah tersimpan di dalam sistem.

Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengobatan Holistik

	<ul style="list-style-type: none"> • Data anamnesis pasien telah tersimpan di dalam sistem. • Data modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit telah tersimpan di dalam sistem.
Post-condition	<ul style="list-style-type: none"> • Sistem berhasil menyimpan data pengobatan holistik. • Aktor Tenaga Medis dapat melihat data pengobatan holistik yang telah berhasil disimpan.
Extension	<ul style="list-style-type: none"> • <i>Extension use case Membaca Panduan Cara Menghindari dan Mengatasi Penyakit</i> pada {membaca modul} pada <i>basic flow</i>. • <i>Extension use case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien</i> pada {memuat detail data pengobatan holistik} pada <i>alternative flow</i>.
Basic Flow	<p>{use case dimulai}</p> <p>{melihat seluruh data pengobatan holistik}</p> <ol style="list-style-type: none"> 1. Sistem menampilkan data pengobatan holistik harian. {menampilkan data pengobatan holistik harian} 2. Aktor memilih satu data pendaftaran pengobatan pasien yang ingin ditambahkan {memilih data pendaftaran pengobatan pasien} 3. Sistem memuat data pendaftaran pengobatan pasien yang telah dipilih oleh aktor Tenaga Paramedis. {memuat data pendaftaran pengobatan terpilih} 4. Sistem memuat data modul penyakit, modul faktor pemicu penyakit, dan modul faktor risiko penyakit. {memuat data modul} 5. Sistem menampilkan formulir untuk menambahkan data pengobatan holistik. 6. Aktor Tenaga Medis menambahkan data penyakit pasien. {membaca modul} 7. Aktor Tenaga Medis menambahkan data modul penyakit. 8. Aktor Tenaga Medis menambahkan data terapi yang diberikan kepada pasien. 9. Aktor Tenaga Medis menambahkan data lokasi intervensi. 10. Aktor Tenaga Medis menambahkan data faktor risiko penyakit dan faktor pemicu penyakit.

Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengobatan Holistik

	<p>11. Aktor Tenaga Medis menambahkan data modul faktor risiko penyakit dan faktor pemicu penyakit.</p> <p>12. Aktor Tenaga Medis memilih untuk menyimpan data pengobatan holistik.</p> <p>13. Sistem melakukan proses penyimpanan data pengobatan holistik.</p> <p>14. Sistem menampilkan pesan bahwa penyimpanan data pengobatan holistik berhasil dilakukan. {menyimpan data pengobatan holistik} {memberi intervensi kesehatan} {use case selesai}</p> <p>15. <i>Use case</i> selesai.</p>
Alternative Flow	<p>A1. Melihat Data Pengobatan Holistik Harian.</p> <p>Jika aktor Tenaga Medis ingin melihat data pengobatan harian, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih untuk melihat data pengobatan holistik harian. 2. <i>Use case</i> dilanjutkan pada {melihat seluruh data pengobatan holistik}. 3. <i>Use case</i> selesai pada {menampilkan data pengobatan holistik harian}. <p>A2. Melihat Seluruh Data Pengobatan Holistik.</p> <p>Pada {melihat seluruh data pengobatan holistik} pada <i>basic flow</i>, jika aktor Tenaga Medis ingin melihat seluruh data pengobatan holistik, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih untuk melihat seluruh data pengobatan holistik. 4. Sistem memuat seluruh data pengobatan holistik. {memuat seluruh data pengobatan holistik} 5. Sistem menampilkan seluruh data pengobatan holistik. {menampilkan seluruh data pengobatan holistik} 6. <i>Use case</i> selesai. <p>A3. Melihat Detail Data Pengobatan Holistik.</p> <p>Pada {menampilkan data pengobatan holistik harian} di <i>basic flow</i>, pada {menampilkan seluruh data pengobatan</p>

Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengobatan Holistik

	<p>holistik} di <i>alternative flow</i>, jika aktor ingin melihat detail data pengobatan holistik, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih satu data pengobatan holistik untuk ditampilkan detailnya. 2. Sistem melakukan proses pemuatan detail data pengobatan holistik. {memuat detail data pengobatan holistik} 3. Sistem menampilkan detail data pengobatan holistik. 4. <i>Use case</i> selesai. <p>A4. Menghapus Data Pengobatan Holistik.</p> <p>Pada {menampilkan data pengobatan holistik harian} di <i>basic flow</i>, pada {menampilkan seluruh data pengobatan holistik} di <i>alternative flow</i>, jika aktor Tenaga Medis ingin menghapus data pengobatan holistik, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih satu data pengobatan holistik untuk dihapus. 2. Sistem melakukan proses penghapusan data pengobatan holistik. {menghapus data pengobatan holistik} 3. Sistem menampilkan pesan bahwa proses penghapusan data pengobatan holistik berhasil dilakukan. 4. <i>Use case</i> selesai. <p>A5. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data pendaftaran pengobatan terpilih} di <i>basic flow</i>, pada {memuat seluruh data pengobatan holistik}, {memuat detail data pengobatan holistik} di <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses pemuatan data, maka sistem menampilkan pesan bahwa pemuatan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A6. Menangani Kondisi Ketika Data Anamnesis Belum Ditambahkan.</p> <p>Pada {memuat data pendaftaran pengobatan terpilih} pada <i>basic flow</i>, jika data anamnesis belum tersedia, maka sistem menampilkan pesan bahwa data anamnesis belum ditambahkan, kemudian <i>use case</i> selesai.</p> <p>A7. Menangani Kegagalan Penyimpanan Data.</p>
--	--

Tabel 4.16 Spesifikasi Use Case Mengelola Data Pengobatan Holistik

	<p>Pada {menyimpan data pengobatan holistik} pada <i>basic flow</i>, jika sistem mengalami kegagalan dalam penyimpanan data pengobatan holistik, maka sistem menampilkan pesan bahwa proses penyimpanan gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A8. Menangani Kegagalan Penghapusan Data.</p> <p>Pada {menghapus data pengobatan holistik} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penghapusan data pengobatan holistik, maka sistem menampilkan pesan bahwa penghapusan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A9. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data pendaftaran pengobatan terpilih}, {memuat data modul} di <i>basic flow</i>, pada {memuat seluruh data pengobatan holistik}, {memuat detail data pengobatan holistik} di <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses pemuatan data, maka sistem menampilkan pesan bahwa proses pemuatan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A10. Menangani Kondisi Ketika Data Pengobatan Holistik Tidak Tersedia.</p> <p>Pada {melihat seluruh data pengobatan holistik} di <i>basic flow</i>, pada {memuat seluruh data pengobatan holistik} di <i>alternative flow</i>, jika data pengobatan holistik tidak tersedia, maka sistem menampilkan pesan bahwa data pengobatan holistik belum tersedia, kemudian <i>use case</i> selesai.</p> <p>A11. Menyimpan Resep Obat.</p> <p>Pada {memberi intervensi kesehatan} pada <i>basic flow</i>, jika aktor Tenaga Medis ingin menyimpan resep obat yang diberikan kepada pasien, maka lakukan <i>subflow Menyimpan Data Resep Obat</i>.</p> <p>A12. Menyimpan Rujukan.</p> <p>Pada {memberi intervensi kesehatan} pada <i>basic flow</i>, jika aktor Tenaga Medis ingin menyimpan rujukan yang diberikan kepada pasien, maka lakukan <i>subflow Menyimpan Data Rujukan</i>.</p> <p>A13. Menyimpan Surat Pengantar Cek Darah.</p> <p>Pada {memberi intervensi kesehatan} pada <i>basic flow</i>, jika aktor Tenaga Medis ingin menyimpan surat pengantar cek</p>
--	--

Tabel 4.16 Spesifikasi *Use Case* Mengelola Data Pengobatan Holistik

	darah yang diberikan kepada pasien, maka lakukan <i>subflow</i> Menyimpan Data Surat Pengantar Cek Darah.
Subflow	<p>S1. Menyimpan Data Resep Obat.</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih untuk menyimpan pemberian resep obat. 2. Sistem melakukan proses penyimpanan data resep obat. {memproses penyimpanan data resep obat} 3. Sistem menampilkan pesan bahwa proses penyimpanan data resep obat berhasil. 4. <i>Use case</i> selesai. <p>S2. Menyimpan Data Rujukan.</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih untuk menambahkan data rujukan. 2. Sistem menampilkan formulir rujukan. 3. Aktor Tenaga Medis memasukkan data rujukan. 4. Aktor Tenaga medis memilih untuk menyimpan data rujukan. 5. Sistem melakukan proses penyimpanan data rujukan. {memproses penyimpanan data rujukan} 6. Sistem menampilkan pesan bahwa proses penyimpanan data rujukan berhasil. 7. <i>Use case</i> selesai. <p>S3. Menyimpan Data Surat Pengantar Cek Darah.</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih untuk menambahkan data surat pengantar cek darah. 2. Sistem melakukan proses penyimpanan data surat pengantar cek darah. {memproses penyimpanan data surat pengantar cek darah} 3. Sistem menampilkan pesan bahwa penyimpanan data surat pengantar cek darah berhasil dilakukan. 4. <i>Use case</i> selesai.

4.3.3.6 Spesifikasi Use Case Membaca Panduan Menghindari dan Mengatasi Penyakit

Tabel 4.17 Spesifikasi Use Case Membaca Panduan Menghindari dan Mengatasi Penyakit

Brief Description	Use case Membaca Panduan Menghindari dan Mengatasi Penyakit menjelaskan bagaimana aktor Tenaga Medis menggunakan sistem untuk mengakses dan membaca modul penyakit, modul faktor risiko penyakit, modul faktor pemicu penyakit yang tersimpan di dalam sistem.
Actor	Tenaga Medis.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan server. • Login yang dilakukan aktor Tenaga Medis berhasil. • Sesi login aktor Tenaga Medis tersimpan dalam sistem. • Aktor Tenaga Medis memilih untuk menambahkan data diagnosis, data faktor risiko penyakit, dan data faktor pemicu penyakit. • Data modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit telah tersimpan di dalam sistem. • Dokumen modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit telah tersimpan di dalam sistem. • Sistem berhasil melakukan proses pemuatan seluruh data modul penyakit, modul faktor risiko penyakit, dan modul faktor pemicu penyakit agar aktor Tenaga Medis dapat memilih data modul tertentu untuk dibaca.
Post-condition	Aktor Tenaga Medis dapat membaca modul penyakit atau modul faktor risiko penyakit atau modul faktor pemicu penyakit.
Basic Flow	<p>{use case dimulai}</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Medis memilih modul penyakit yang ingin dibaca. 2. Sistem memuat modul penyakit yang sudah dipilih oleh aktor Tenaga Medis. <p>{memuat modul penyakit}</p> <ol style="list-style-type: none"> 3. Aktor Tenaga Medis memilih untuk membaca modul penyakit yang telah dipilih.

Tabel 4.17 Spesifikasi Use Case Membaca Panduan Menghindari dan Mengatasi Penyakit

	<p>4. Sistem menampilkan dokumen modul penyakit yang telah dipilih. {menampilkan modul penyakit}</p> <p>5. Aktor Tenaga Medis memilih modul faktor risiko penyakit yang ingin dibaca.</p> <p>6. Sistem memuat modul faktor risiko penyakit yang sudah dipilih oleh aktor Tenaga Medis. {memuat modul faktor risiko penyakit}</p> <p>7. Aktor Tenaga Medis memilih untuk membaca modul faktor risiko penyakit yang telah dipilih.</p> <p>8. Sistem menampilkan dokumen modul faktor risiko penyakit yang telah dipilih. {menampilkan modul faktor risiko penyakit}</p> <p>9. Aktor Tenaga Medis memilih modul faktor pemicu penyakit yang ingin dibaca.</p> <p>10. Sistem memuat modul faktor pemicu penyakit yang sudah dipilih oleh aktor Tenaga Medis. {memuat modul faktor pemicu penyakit}</p> <p>11. Aktor Tenaga Medis memilih untuk membaca modul faktor pemicu penyakit yang telah dipilih.</p> <p>12. Sistem menampilkan dokumen modul faktor penyakit yang telah dipilih. {menampilkan modul faktor pemicu penyakit} {use case selesai}</p> <p>13. <i>Use case</i> selesai.</p>
Alternative Flow	<p>A1. Menangani Kegagalan Pemuatan Data Modul.</p> <p>Pada {memuat modul penyakit}, {memuat modul faktor risiko penyakit}, {memuat modul faktor pemicu penyakit} pada <i>basic flow</i>, jika sistem mengalami kegagalan dalam proses pemuatan data modul, maka sistem menampilkan pesan bahwa proses pemuatan data modul gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A2. Menangani Kondisi Ketika Dokumen Modul Tidak Tersedia.</p> <p>Pada {menampilkan modul penyakit}, {menampilkan modul faktor risiko penyakit}, {menampilkan modul faktor pemicu penyakit} pada <i>basic flow</i>, jika data modul tidak tersedia di dalam sistem, maka sistem menampilkan pesan bahwa</p>

Tabel 4.17 Spesifikasi Use Case Membaca Panduan Menghindari dan Mengatasi Penyakit

	dokumen modul tidak dapat ditemukan, kemudian <i>use case</i> selesai.
--	--

4.3.3.7 Spesifikasi Use Case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien

Tabel 4.18 Spesifikasi Use Case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien

Brief Description	<i>Use case</i> Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien menunjukkan bagaimana aktor Tenaga Medis menggunakan sistem untuk melihat hasil perhitungan tingkat risiko penyakit dan tingkat stres pasien berdasarkan data dasar kesehatan keluarga yang telah tersimpan di dalam sistem.
Actor	Tenaga Medis
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Tenaga Medis berhasil. • Sesi <i>login</i> aktor Tenaga Medis tersimpan dalam sistem. • Sistem berhasil menampilkan data dasar kesehatan keluarga.
Post-condition	Aktor Tenaga Medis mengetahui hasil perhitungan tingkat risiko penyakit dan tingkat stres pasien berdasarkan data dasar kesehatan keluarga yang tersimpan di dalam sistem.
Basic Flow	<p>{use case dimulai}</p> <ol style="list-style-type: none"> 1. Sistem memuat hasil perhitungan tingkat risiko penyakit dan tingkat stres pasien. {memuat hasil perhitungan} 2. Sistem menampilkan tingkat risiko penyakit dan tingkat stres pasien. {use case selesai} 3. <i>Use case</i> selesai.
Alternative Flow	<p>A1. Menangani Kegagalan Pemuatan Hasil Perhitungan.</p> <p>Pada {memuat hasil perhitungan} pada <i>basic flow</i>, jika sistem mengalami kegagalan pemuatan hasil perhitungan tingkat risiko penyakit dan tingkat stres pasien, maka sistem menampilkan pesan bahwa hasil perhitungan tingkat risiko</p>

Tabel 4.18 Spesifikasi Use Case Melihat Tingkat Risiko Penyakit dan Tingkat Stres Pasien

	penyakit dan tingkat stres pasien belum tersedia, kemudian <i>use case</i> selesai.
--	---

4.3.3.8 Spesifikasi Use Case Mengelola Data Persediaan Obat

Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat

Brief Description	<i>Use case</i> Mengelola Data Persediaan Obat menunjukkan bagaimana aktor Tenaga Kefarmasian menggunakan sistem untuk melakukan pencatatan obat keluar, pencatatan obat masuk, menambahkan data obat, dan mencetak laporan obat keluar.
Actor	Tenaga Kefarmasian.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Tenaga Kefarmasian berhasil. • Sesi <i>login</i> aktor Tenaga Kefarmasian tersimpan dalam sistem. • Seluruh data obat poliklinik telah tersimpan di dalam sistem.
Post-condition	Aktor Tenaga Kefarmasian dapat melihat data persediaan obat yang tersimpan di dalam sistem.
Basic Flow	<p>{use case dimulai}</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk mencatat obat keluar. 2. Sistem memuat data obat poliklinik. {memuat data obat poliklinik} 3. Sistem menampilkan data obat poliklinik. 4. Aktor Tenaga Kefarmasian memasukkan data obat keluar. 5. Aktor Tenaga Kefarmasian memilih untuk menyimpan pencatatan obat keluar. 6. Sistem melakukan proses penyimpanan data obat keluar. {menyimpan data obat keluar} 7. Sistem menampilkan pesan bahwa proses penyimpanan data obat keluar berhasil dilakukan {use case selesai}. 8. <i>Use case</i> selesai.

Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat

<p>Alternative Flow</p>	<p>A1. Melihat Seluruh Data Obat.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Tenaga Kefarmasian ingin melihat seluruh data obat, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk melihat seluruh data obat. 2. Lakukan <i>subflow</i> Melihat Seluruh Data Obat. <p>A2. Melihat Data Obat Masuk.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Tenaga Kefarmasian ingin melihat data obat masuk, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk melihat data obat masuk. 2. Lakukan <i>subflow</i> Melihat Data Obat Masuk. <p>A3. Melihat Data Obat Keluar.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Tenaga Kefarmasian ingin melihat data obat keluar, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk melihat data obat keluar. 2. Lakukan <i>subflow</i> Melihat Data Obat Keluar.. <p>A4. Mengeluarkan Laporan Obat Keluar.</p> <p>Pada {menampilkan data obat keluar} pada <i>subflow</i>, jika aktor Tenaga Kefarmasian ingin mengeluarkan data laporan obat, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk mengeluarkan laporan obat keluar. 2. Sistem melakukan proses pengeluaran laporan obat keluar dalam dokumen PDF. {mengeluarkan laporan obat keluar} 3. <i>Use case</i> selesai. <p>A5. Menyimpan Data Obat Baru.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Tenaga Kefarmasian memilih untuk menambah data obat baru, maka:</p> <ol style="list-style-type: none"> 1. Aktor Tenaga Kefarmasian memilih untuk menambah data obat baru. 2. Aktor Tenaga Kefarmasian memasukkan data obat baru.
--------------------------------	--

Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat

	<ol style="list-style-type: none"> Aktor Tenaga Kefarmasian memilih untuk menyimpan data obat baru. Sistem melakukan proses penyimpanan data obat baru. {menyimpan data obat baru} Sistem menampilkan pesan bahwa data obat baru berhasil disimpan. Lakukan <i>subflow</i> Melihat Seluruh Data Obat. <p>A6. Menyimpan Pencatatan Data Obat Masuk.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Tenaga Kefarmasian memilih untuk melakukan pencatatan data obat masuk, maka:</p> <ol style="list-style-type: none"> Aktor Tenaga Kefarmasian memilih untuk mencatat data obat masuk. Aktor Tenaga Kefarmasian memasukkan data obat masuk. Aktor Tenaga Kefarmasian memilih untuk menyimpan catatan data obat masuk. Sistem melakukan proses penyimpanan catatan data obat masuk. {menyimpan catatan data masuk} Sistem menampilkan pesan bahwa pencatatan obat masuk berhasil dilakukan. Lakukan <i>subflow</i> Melihat Data Obat Masuk. <p>A7. Menangani Kegagalan Penambahan Data Obat Baru.</p> <p>Pada {menyimpan data obat baru} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penyimpanan data obat baru, maka sistem akan menampilkan pesan bahwa proses penambahan data obat baru gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A8. Menangani Kegagalan Pencatatan Obat Masuk.</p> <p>Pada {menyimpan data obat baru} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penyimpanan data obat baru, maka sistem akan menampilkan pesan bahwa proses penambahan data obat baru gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A9. Menangani Kondisi Ketika Data Obat Poliklinik Belum Tersedia.</p>
--	---

Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat

	<p>Pada {memuat data obat poliklinik} pada <i>basic flow</i>, jika data obat poliklinik belum tersedia, maka sistem menampilkan pesan bahwa data obat belum tersedia, kemudian <i>use case</i> selesai.</p> <p>A10. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data obat poliklinik} pada <i>basic flow</i>, pada {memuat seluruh data obat}, {memuat data obat masuk}, {memuat data obat keluar} pada <i>subflow</i>, jika sistem mengalami kegagalan dalam proses pemuatan data, maka sistem menampilkan pesan bahwa proses pemuatan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A11. Menangani Kegagalan Pencatatan Obat Keluar.</p> <p>Pada {menyimpan data obat keluar} pada <i>basic flow</i>, jika sistem mengalami kegagalan dalam proses penyimpanan data obat keluar, maka sistem menampilkan pesan bahwa proses penyimpanan data obat keluar gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A12. Menangani Kegagalan Pengeluaran Laporan Obat Keluar.</p> <p>Pada {mengeluarkan laporan obat keluar} pada <i>alternative flow</i>, jika sistem mengalami kegaalan dalam proses pengeluaran laporan obat keluar, maka sistem menampilkan pesan bahwa proses pengeluaran laporan data obat keluar gagal dilakukan, kemudian <i>use case</i> selesai.</p>
Subflow	<p>S1. Melihat Seluruh Data Obat.</p> <ol style="list-style-type: none"> 1. Sistem melakukan proses pemuatan seluruh data obat. {memuat seluruh data obat} 2. Sistem menampilkan seluruh data obat. 3. <i>Use case</i> selesai. <p>S2. Melihat Data Obat Masuk.</p> <ol style="list-style-type: none"> 1. Sistem melakukan proses pemuatan data obat masuk. {memuat data obat masuk} 2. Sistem menampilkan data obat masuk. 3. <i>Use case</i> selesai. <p>S3. Melihat Data Obat Keluar.</p> <ol style="list-style-type: none"> 1. Sistem melakukan proses pemuatan data obat keluar. {memuat data obat keluar}

Tabel 4.19 Spesifikasi Use Case Mengelola Data Persediaan Obat

	<ol style="list-style-type: none"> 2. Sistem menampilkan data obat keluar. {menampilkan data obat keluar} 3. <i>Use case</i> selesai.
--	--

4.3.3.9 Spesifikasi Use Case Mengelola Data Wilayah

Tabel 4.20 Spesifikasi Use Case Mengelola Data Wilayah

Brief Description	<i>Use Case</i> Mengelola Data Wilayah menunjukkan bagaimana aktor Admin Sistem menggunakan sistem untuk menambah, mengubah, dan menghapus data wilayah administratif Indonesia.
Actor	Admin Sistem.
Pre-condition	<ul style="list-style-type: none"> • Komputer yang digunakan untuk mengakses sistem terhubung dengan internet. • Sistem terhubung dengan <i>server</i>. • <i>Login</i> yang dilakukan aktor Admin Sistem berhasil. • Sesi <i>login</i> aktor Admin Sistem tersimpan dalam sistem.
Post-condition	Aktor Admin Sistem dapat melihat hasil perubahan data wilayah administratif Indonesia yang tersimpan di dalam sistem.
Basic Flow	<p>{use case dimulai}</p> <ol style="list-style-type: none"> 1. Aktor Admin Sistem memilih untuk menambah data wilayah. 2. Sistem menampilkan formulir penambahan data wilayah. 3. Aktor Admin Sistem memasukkan data wilayah. 4. Aktor Admin Sistem memilih untuk menyimpan data wilayah. 5. Sistem melakukan proses penyimpanan data wilayah. {menambah data wilayah} 6. Sistem menampilkan pesan bahwa proses penambahan data wilayah berhasil dilakukan. 7. Lakukan <i>subflow</i> Menampilkan Seluruh Data Wilayah.
Alternative Flow	<p>A1. Melihat Seluruh Data Wilayah.</p> <p>Pada {use case dimulai} pada <i>basic flow</i>, jika aktor Admin Sistem ingin melihat seluruh data wilayah, maka:</p>

Tabel 4.20 Spesifikasi Use Case Mengelola Data Wilayah

	<ol style="list-style-type: none"> 1. Aktor Admin Sistem memilih untuk melihat seluruh data wilayah. 2. Lakukan <i>subflow</i> Menampilkan Seluruh Data Wilayah. <p>A2. Mengubah Data Wilayah.</p> <p>Pada {menampilkan seluruh data wilayah} pada <i>subflow</i>, jika aktor Admin Sistem ingin mengubah data wilayah, maka:</p> <ol style="list-style-type: none"> 1. Aktor memilih satu data wilayah yang ingin diubah. 2. Sistem melakukan proses pemuatan data wilayah yang telah dipilih. {memuat data wilayah untuk diubah} 3. Sistem menampilkan data wilayah yang telah dipilih 4. Aktor Admin Sistem mengubah data wilayah. 5. Aktor Admin Sistem memilih untuk menyimpan perubahan data wilayah. 6. Sistem melakukan proses penyimpanan perubahan data wilayah. {mengubah data wilayah} 7. Sistem menampilkan pesan bahwa proses pengubahan data wilayah berhasil dilakukan. 8. Lakukan <i>subflow</i> Menampilkan Seluruh Data Wilayah. 9. <i>Use case</i> selesai. <p>A3. Menghapus Data Wilayah.</p> <p>Pada {menampilkan seluruh data wilayah} pada <i>subflow</i>, jika aktor Admin Sistem ingin menghapus data wilayah, maka:</p> <ol style="list-style-type: none"> 1. Aktor memilih satu data wilayah yang ingin dihapus. 2. Sistem melakukan proses penghapusan data wilayah yang telah dipilih. {menghapus data wilayah} 3. Sistem menampilkan pesan bahwa proses penghapusan data wilayah berhasil dilakukan. 4. Lakukan <i>subflow</i> Menampilkan Seluruh Data Wilayah. 5. <i>Use case</i> selesai. <p>A4. Menangani Kegagalan Proses Penambahan Data Wilayah.</p>
--	---

Tabel 4.20 Spesifikasi Use Case Mengelola Data Wilayah

	<p>Pada {menambah data wilayah} pada <i>basic flow</i>, jika sistem mengalami kegagalan dalam proses penyimpanan data wilayah, maka sistem menampilkan pesan bahwa proses penyimpanan data wilayah gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A5. Menangani Kegagalan Proses Pengubahan Data Wilayah.</p> <p>Pada {mengubah data wilayah} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses pengubahan data wilayah, maka sistem menampilkan pesan bahwa proses pengubahan data wilayah gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A6. Menangani Kegagalan Proses Penghapusan Data Wilayah.</p> <p>Pada {menghapus data wilayah} pada <i>alternative flow</i>, jika sistem mengalami kegagalan dalam proses penghapusan data, maka sistem menampilkan pesan bahwa proses penghapusan data gagal dilakukan, kemudian <i>use case</i> selesai.</p> <p>A7. Menangani Kegagalan Pemuatan Data.</p> <p>Pada {memuat data wilayah untuk diubah} di <i>alternative flow</i>, pada {memuat seluruh data wilayah} di <i>subflow</i>, jika sistem mengalami kegagalan dalam proses pemuatan data, maka sistem menampilkan pesan bahwa pemuatan data gagal dilakukan, kemudian <i>use case</i> selesai.</p>
Subflow	<p>S1. Menampilkan Seluruh Data Wilayah.</p> <ol style="list-style-type: none"> 1. Sistem melakukan proses pemuatan data wilayah {memuat seluruh data wilayah} 2. Sistem menampilkan data wilayah. {menampilkan seluruh data wilayah} 3. <i>Use case</i> selesai.

4.4 Activity Model

Salah satu cara memvisualisasi *flow of events* yang dimiliki oleh *use-case* adalah dengan menggunakan UML activity diagram (Bittner & Spence, 2002). Bittner dan Spence menambahkan, teknik visualisasi ini memberikan kemudahan kepada *stakeholder* untuk memahami *flow of events* (*basic flows* dan *alternative flows*), sehingga *stakeholder* dapat memahami cakupan *use-case*. Activity diagram dalam penelitian ini dibuat berdasarkan *flow of events* dari masing-masing *use-case* pada sub-bab sebelumnya.

DAFTAR PUSTAKA

- 1keydata, 2009. *Physical Data Model*. [Online] Available at: <http://www.1keydata.com/datawarehousing/physical-data-model.html> [Diakses 3 September 2017].
- Aggarwal, V., 2002. The Application of the Unified Modeling Language in Object-Oriented Analysis of Healthcare Information Systems. *Journal of Medical Systems*, 26(5), pp. 383-397.
- Ambyssoft Inc., 2013. *Physical Data Model (PDM)s: An Agile Introduction*. [Online] Available at: <http://agilemodeling.com/artifacts/physicalDataModel.htm> [Diakses 3 September 2017].
- Becker, J., Kugeler, M. & Rosemann, M., 2013. *Process Management: A Guide for the Design of Business Processes*. Berlin: Springer-Verlag.
- Bittner, K. & Spence, I., 2002. *Use Case Modeling*. Boston: Addison Wesley.
- Booch, G., Rumbaugh, J. & Jacobson, I., 1998. *The Unified Modeling Language User Guide*. Boston: Addison Wesley.
- Departemen Kesehatan RI, 2009. *Departemen Kesehatan Republik Indonesia*. [Online] Available at: http://pppl.depkes.go.id/asset/regulasi/KEPMENKES_374-2009_TTG_SKN-2009.pdf [Diakses 3 4 2017].
- IBM Knowledge Center, 2014. *IBM Knowledge Center*. [Online] Available at: https://www.ibm.com/support/knowledgecenter/en/SS6RBX_11.4.3/com.ibm.sa.bpr.doc/topics/t_Data_obj.html [Diakses 3 September 2017].
- Krol, M. & Reich, D. L., 1999. Object-Oriented Analysis and Design of a Health Care Management Information System. *Journal of Medical Systems*, 24(2), pp. 145-158.
- Krol, P. & Kruchten, P., 2003. *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Boston: Addison Wesley.
- Leffingwell, D. & Widrig, D., 2003. *Managing Software Requirements: A Use Case Approach*. 2nd penyunt. Boston: Addison Wesley.
- Monk, E. & Wagner, B., 2013. *Concepts in Enterprise Resource Planning*. 4th penyunt. Boston: Course Technology.
- Nugraha, D. C. A., 2017. *A Business Architecture Modeling Methodology to Support the Integration of Primary Health Care: Implementation of Primary Health Care in Indonesia*. s.l.:s.n.

Object Management Group, 2005. *OMG Unified Modeling Language Specification*. [Online]

Available at: <http://www.omg.org/spec/UML/2.0/> [Diakses 31 August 2017].

Object Management Group, 2005. *Unified Modeling Language: Superstructure*. [Online]

Available at: <http://www.omg.org/spec/UML/2.0/Superstructure/PDF/> [Diakses 31 August 2017].

Object Management Group, 2011. *Business Process Model and Notation (BPMN)*. [Online]

Available at: <http://www.omg.org/spec/BPMN/2.0/PDF> [Diakses 20 Juni 2017].

Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach*. 7th penyunt. New York: McGraw-Hill.

Przybyłek, A., 2011. *Bridging the gap between business process models and use-case models*. Gdańsk, Gdańsk University of Technology.

Roebijoso, J., 2010. Model Manajemen Strategik dan Operasional, Pelayanan Kesehatan Dokter Keluarga yang Berwawasan Mutu pada Karyawan dan Keluarga.

Rumbaugh, J., Jacobson, I. & Booch, G., 2004. *The Unified Modeling Language Reference Manual*. 2nd penyunt. Boston: Addison Wesley.

Sommerville, I., 2011. *Software Engineering*. 9th penyunt. Boston: Addison-Wesley.