



TUGAS AKHIR - KS141501

**ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS
WEBSITE DAN MOBILE ANDROID
STUDI KASUS : RSIA PONDOK TJANDRA**

***ANALYSIS AND DESIGN “ANTRI” WEBSITE BASED
AND MOBILE ANDROID
CASE STUDY : RSIA PONDOK TJANDRA***

**NURUL FAKHRIA
NRP 5209 100 097**

**Dosen Pembimbing
Feby Artwodini Muqtadiroh, S.Kom., M.T.**

**JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017**



TUGAS AKHIR - KS141501

**ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS
WEBSITE DAN MOBILE ANDROID
STUDI KASUS : RSIA PONDOK TJANDRA**

**NURUL FAKHRIA
NRP 5209 100 097**

**Dosen Pembimbing
Feby Artwodini Muqtadiroh, S.Kom., M.T**

**JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017**

FINAL PROJECT – KS 141501

***ANALYSIS AND DESIGN “ANTRI” WEBSITE BASED AND
MOBILE ANDROID
CASE STUDY : RSIA PONDOK TJANDRA***

Nurul Fakhria
5209 100 097

Supervisors

Feby Artwodini Muqtadiroh, S.Kom., M.T.

**INFORMATION SYSTEMS DEPARTMENT
Information Technology Faculty
Sepuluh Nopember Institut of Technology
Surabaya 2017**

LEMBAR PENGESAHAN

**ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS
WEBSITE DAN MOBILE ANDROID
STUDI KASUS : RSIA PONDOK TJANDRA**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

NURUL FAKHRIA
NRP. 5209 100 097

Surabaya, 20 Juli 2017

**KETUA —
JURUSAN SISTEM INFORMASI**

Dr. Ir. Aris Tjahyanto, M.Kom
NIP.19650310 199102 1 001



LEMBAR PERSETUJUAN

**ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS
WEBSITE DAN MOBILE ANDROID
STUDI KASUS : RSIA PONDOK TJANDRA**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

Nurul Fakhria
NRP 5209 100 097

Disetujui Tim Penguji : Tanggal Ujian : 05 Juli 2017
Periode Wisuda : September 2017

Feby Artwodini M., S.Kom., M.T.


(Pembimbing I)

Tony Dwi Susanto, S.T., M.T., Ph.D


(Penguji I)

Eko Wahyu Tyas D., S.Kom, MBA


(Penguji II)

**ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS
WEBSITE DAN MOBILE ANDROID
STUDI KASUS : RSIA PONDOK TJANDRA**

Nama Mahasiswa : Nurul Fakhria
NRP : 5209 100 097
Jurusan : Sistem Informasi FTIf-ITS
Dosen Pembimbing 1 : Feby Artwodini M., S.Kom.,M.T.

ABSTRAK

Antrian dapat dijumpai pada semua aspek kehidupan terutama dalam industri jasa, salah satunya di lingkup rumah sakit. Jumlah pasien yang cukup banyak dan pada saat tertentu, antrian pemeriksaan akan menjadi sangat lama dan hal ini berdampak pada kenyamanan pasien dalam menunggu. Sistem antrian manual yang saat ini digunakan perlu dikembangkan lebih lanjut agar dapat meningkatkan pelayanan, kenyamanan dan kepuasan pasien. Dengan pesatnya kemajuan teknologi, salah satu cara yang bisa dilakukan untuk meningkatkan peningkatan pelayanan dan kepuasan pasien adalah dengan membuat aplikasi antrian berbasis website untuk pihak rumah sakit dan aplikasi berbasis mobile android untuk pasien. Agar mendapatkan hasil yang diharapkan mampu menyelesaikan masalah tersebut, perlu dilakukan analisis dan perancangan aplikasi yang sesuai dengan kebutuhan.

Analisis dan perancangan aplikasi dibuat menggunakan metode pendekatan yang berorientasi objek (OOAD) dengan metode pemodelan Unified Modeling Language (UML). Metode pengembangan sistem yang digunakan adalah metode waterfall.

Hasil akhir dari analisis dan perancangan ini berupa dokumen analisis (Spesifikasi Kebutuhan Perangkat Lunak) dan dokumen desain (Dokumen Desain Perangkat Lunak).

Dokumen tersebut diharapkan bisa menjadi pedoman pengembangan aplikasi ANTRI tersebut.

Kata kunci : Waterfall, Object Oriented Analysis Design, Unified Modelling Language, SKPL, DDPL.

**ANALYSIS AND DESIGN “ANTRI” BASED WEBISTE
AND MOBILE ANDROID
CASE STUDY : RSIA PONDOK TJANDRA**

Name : Nurul Fakhria
NRP : 5209 100 097
Departement : Information System FTIf-ITS
Supervisor 1 : Feby Artwodini M., S.Kom., M.T.

ABSTRACT

Queues can be found in all aspects of life, especially in the service industry, one of which is at the hospital. Many patients and at any given time, queue of doctor examination will be very long and this impact on patient comfort in waiting. The manual queuing system currently in use needs to be further developed in order to improve service, comfort and patient satisfaction. With the rapid advancement of technology, one of the ways that can be done to improve service improvement and patient satisfaction is to create a website-based queue application for hospitals and mobile android based applications for patients. In order to get the expected results are able to solve the problem, need to be analyzed and design the application in accordance with the needs.

Analysis and design of application is made using object-oriented approach (OOAD) method with Unified Modeling Language (UML) modeling. System development method used is waterfall method.

The result of this analysis and design is an analysis document (Software Requirement Specification) and design document (Document Design Software). The document is expected to be a guideline for the development of ANTRI applications.

***Keywords: Waterfall, Object Oriented Analysis Design,
Unified Modeling Language, SKPL, DDPL***

KATA PENGANTAR

Alhamdulillah, puji syukur kepada Allah SWT yang telah memberikan rahmat dan kekuatan kepada penulis sehingga dapat menyelesaikan tugas akhir yang berjudul:

ANALISIS DAN PERANCANGAN “ANTRI” BERBASIS WEBSITE DAN MOBILE ANDROID STUDI KASUS : RSIA PONDOK TJANDRA

Penulis juga mengucapkan terima kasih sedalam-dalamnya kepada semua pihak yang telah memberikan semangat, bimbingan, arahan, serta bantuan dalam menyelesaikan tugas akhir ini

1. Rekan-rekan staf di RSIA Pondok TJandra yang telah membantu penyempurnaan dalam pengumpulan dan pengolahan data.
2. Ibu Feby Artwodini Muqtadiroh, S.Kom., M.T. selaku Dosen Pembimbing yang selalu memberi saran dan arahan sehingga Tugas Akhir ini dapat terselesaikan dengan sebaik mungkin.
3. Bapak Ir. Aris Tjahyanto, S.Kom., M.Kom, selaku ketua jurusan Sistem Informasi ITS
4. Bapak Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom. selaku dosen wali yang telah memberikan masukan kepada penulis pada saat perwalian baik mengenai perkuliahan sampai pengerjaan Tugas Akhir ini.
5. Kedua orang tua, Bapak dan Almh. Ibu tercinta yang selalu memberikan dukungan serta doa yang tidak pernah putus sehingga penulis bisa semakin kuat dalam menyelesaikan Tugas Akhir ini.

6. Kakak-Kakak, Anak dan Suami tersayang yang selalu memberikan dukungan dan doa sampai Tugas Akhir ini selesai.
7. Teman-teman seperjuangan yang sangat memberikan bantuan serta dorongan yang kuat..

Penyusunan laporan ini masih jauh dari sempurna, untuk itu penulis menerima adanya kritik dan saran yang membangun untuk perbaikan di masa mendatang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca maupun bagi peneliti selanjutnya.

Surabaya, Juni 2017

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	iii
LEMBAR PERSETUJUAN... Error! Bookmark not defined.	
ABSTRAK.....	v
ABSTRACT.....	vii
KATA PENGANTAR	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan Tugas Akhir	3
1.5 Manfaat Tugas Akhir	4
1.5.1 Manfaat Akademis.....	4
1.5.2 Manfaat bagi Perusahaan.....	4
1.6 Relevansi.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Studi Sebelumnya	5
2.2 Dasar Teori	6
2.2.1 Requirement Engineering	6
2.2.2 Software Design	8
2.2.3 Object Oriented Analysis and Design (OOAD) ...	10
2.2.4 Unified Modeling Language (UML)	11
2.2.5 Waterfall Model.....	16
BAB III METODOLOGI PENELITIAN.....	19
3.1 Tahapan Persiapan	20
3.1.1 Studi Literatur.....	20
3.2 Tahapan Perancangan Perangkat Lunak	21
3.2.1 Analisis.....	21
3.2.2 Desain.....	22
3.2.3 Validasi.....	23

3.2.3.1 Matriks Kerunutan.....	23
3.3 Tahapan Penyelesaian.....	23
3.3.1 Laporan.....	24
BAB IV PERANCANGAN	25
4.1 Studi Literatur	25
4.2 Perancangan Perangkat Lunak.....	25
4.2.1 Analisis.....	25
4.2.2 Desain.....	28
4.2.3 Validasi.....	31
4.2.3.1 Matriks Kerunutan.....	31
4.3 Tahapan Penyelesaian.....	32
BAB V IMPLEMENTASI.....	33
5.1 Proses Pelaksanaan Penelitian	33
5.1.1 Pengambilan Data.....	33
5.2 Hambatan dan Rintangan dalam Pelaksanaan	34
5.3 Hasil Review Dokumen	34
5.3.1 Rumah Sakit Ibu dan Anak (RSIA) Pondok Tjandra	34
BAB VI HASIL DAN PEMBAHASAN	41
6.1 Perancangan Perangkat Lunak.....	41
6.1.1 Analisis.....	41
6.1.2 Desain.....	54
6.1.3 Validasi.....	58
BAB VII KESIMPULAN DAN SARAN	65
7.1 Kesimpulan.....	65
7.2 Saran	66
BIODATA PENULIS	1-69
LAMPIRAN A – HASIL WAWANCARA	1-71
LAMPIRAN B – WAWANCARA DENGAN PASIEN	1-74
LAMPIRAN C – WAWANCARA KE-DUA.....	1-76
LAMPIRAN D – WAWANCARA KE-TIGA.....	1-77
LAMPIRAN E – SKPL.....	1-79
LAMPIRAN D – DDPL.....	1

DAFTAR GAMBAR

Gambar 2.1 Dimensi RE	7
Gambar 2.2 Use Case Diagram	12
Gambar 2.3 Activity Diagram	13
Gambar 2.4 Sequence Diagram	14
Gambar 2.5 Class Diagram	15
Gambar 2.6 One-to-one relationship	16
Gambar 2.7 One-to-many relationship	16
Gambar 2.8 Waterfall model	17
Gambar 4.1 Contoh Use-case Diagram	29
Gambar 4.2 Contoh Sequence Diagram	29
Gambar 4.3 Contoh Class Diagram	30
Gambar 4.4 Contoh Desain Antarmuka (GUI)	31
Gambar 5.1 Struktur Organisasi RSIA Pondok Tjandra	37
Gambar 5.2 alur pelayanan IGD	38
Gambar 5.3 alur pelayanan rawat jalan	39
Gambar 5.4 Alur pelayanan rawat inap	40
Gambar 6.1 Flowchart Pelayanan RS	44
Gambar 6.2 Flowchart sistem baru untuk pasien baru	45
Gambar 6.3 Flowchart sistem baru untuk pasien lama	46
Gambar 6.4 Use-Case Diagram Administrator	54
Gambar 6.5 Sequence Input Data Dokter	55
Gambar 6.6 Class Diagram Admin	56
Gambar 6.7 user interface Login website	56
Gambar 6.8 User Interface Tambah Data Pasien	57
Gambar 6.9 UI Jadwal Dokter Android	57
Gambar 6.10 UI Login Android	57
Gambar 6.11 Design Arsitektur Aplikasi	58

DAFTAR TABEL

Tabel 2.1 Studi Sebelumnya.....	5
Tabel 3.1 Metode Penelitian.....	19
Tabel 4.1 Contoh Matriks Kerunutan.....	31
Tabel 4.2 Kebutuhan Fungsional	48
Tabel 4.3 Usability Requirement	49
Tabel 4.4 Reliability Requirement	49
Tabel 4.5 Portability Requirement	50
Tabel 4.6 Supportability Requirement	50
Tabel 4.7 Use-Case	50
Tabel 4.8 Use-Case (UCD) Ubah Data Dokter	52
Tabel 4.9 Mariks Kerunutan.....	60

BAB I

PENDAHULUAN

Pada bab ini akan menjelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah dan tujuan penelitian yang menjadi dasar pelaksanaan penelitian tugas akhir dan penjelasan sistematika penulisan laporan tugas akhir ini.

1.1 Latar Belakang

Sistem informasi telah menjadi bagian yang tak terpisahkan dari kegiatan bisnis suatu perusahaan atau organisasi modern. Sistem informasi merupakan aplikasi yang sangat dibutuhkan dan diminati. Untuk menghasilkan sistem informasi yang bisa membantu dan mendukung kegiatan bisnis dan manajemen perusahaan atau organisasi tidaklah mudah. Ada langkah-langkah dan metode terstandarisasi yang harus diikuti untuk menghasilkan sistem informasi yang baik. Langkah-langkah standar inilah yang menghasilkan tahapan-tahapan, yaitu analisis dan perancangan sistem informasi yang berisi serangkaian langkah dan dokumentasi baku yang harus diikuti oleh pengembang sistem informasi untuk menjamin diperolehnya sistem informasi yang bisa menyelesaikan permasalahan-permasalahan bisnis, sekaligus mempermudah pengembangan sistem di masa depan.

Menurut WHO (*World Health Organization*), rumah sakit adalah bagian integral dari suatu organisasi sosial dan kesehatan dengan fungsi menyediakan pelayanan paripurna (komprehensif), penyembuhan penyakit (kuratif) dan pencegahan penyakit (preventif) kepada masyarakat. Rumah sakit juga merupakan pusat pelatihan bagi tenaga kesehatan dan pusat penelitian medik [1]. Berdasarkan undang-undang No. 44 Tahun 2009 tentang rumah sakit, yang dimaksudkan dengan rumah sakit adalah institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan secara

paripurna yang menyediakan pelayanan rawat inap, rawat jalan, dan gawat darurat [2].

Antrian dapat dijumpai pada semua aspek kehidupan terutama dalam industri jasa, salah satunya di lingkup rumah sakit. Jumlah pasien yang cukup banyak dan pada saat tertentu, antrian pemeriksaan akan menjadi sangat lama dan hal ini berdampak pada kenyamanan pasien dalam menunggu. Sistem antrian manual yang saat ini digunakan perlu dikembangkan lebih lanjut agar dapat meningkatkan pelayanan, kenyamanan dan kepuasan pasien. Dengan pesatnya kemajuan teknologi, salah satu cara yang bisa dilakukan untuk meningkatkan peningkatan pelayanan dan kepuasan pasien adalah dengan membuat aplikasi antrian berbasis website untuk pihak rumah sakit dan aplikasi berbasis mobile android untuk pasien. Agar mendapatkan hasil yang diharapkan mampu menyelesaikan masalah tersebut, perlu dilakukan analisis dan perancangan aplikasi yang sesuai dengan kebutuhan.

Rumah sakit ibu dan anak(RSIA) Pondok Tjandra dalam melayani pasien masih secara manual. Apapun kegiatannya, seperti pendaftaran pemeriksaan pasien, pelayanan obat di apotek, administrasi dan kegiatan lainnya. Studi kasus yang saya ambil disini adalah tentang pendaftaran pemeriksaan pasien beserta antriannya. Pendaftaran pemeriksaan yang dilakukan masih manual bisa melalui telepon dan kemudian dilakukan pencataan oleh petugas lalu di beri nomer antrian atau dengan cara langsung datang ketika jadwal dokter praktek. Oleh sebab itu, banyak pasien yang merasa kurang puas dengan pelayanannya.

Dengan adanya sistem ini, diharapkan dapat membantu memperbaiki pelayanan terhadap pasien. Karena rumah sakit menjual jasa, jika pelayanan kurang memuaskan, maka pasien akan berkurang dan menyebabkan pendapatan menurun. Agar mendapatkan hasil yang diharapkan mampu menyelesaikan

masalah tersebut, perlu dilakukan analisis dan perancangan sistem yang sesuai dengan kebutuhannya.

Analisis dan perancangan aplikasi dibuat menggunakan metode pendekatan yang berorientasi objek (OOAD) dengan metode pemodelan Unified Modeling Language (UML). Metode pengembangan sistem yang digunakan adalah metode waterfall. Hasil akhir dari analisis dan perancangan ini berupa dokumen analisis (Spesifikasi Kebutuhan Perangkat Lunak) dan dokumen desain (Dokumen Desain Perangkat Lunak). Dokumen tersebut diharapkan bisa menjadi pedoman pengembangan aplikasi ANTRI tersebut.

1.2 Rumusan Masalah

Dari uraian latar belakang di atas, dapat disusun rumusan masalah sebagai berikut:

1. Bagaimana kebutuhan aplikasi ANTRI pada RSIA Pondok Tjandra?
2. Bagaimana perancangan/desain aplikasi ANTRI pada RSIA pondok Tjandra agar sesuai dengan kebutuhan yang telah didefinisikan?

1.3 Batasan Masalah

Dari rumusan masalah yang tersusun di atas, batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi ANTRI hanya untuk mengambil nomer antrian pemeriksaan, mengetahui waktu dan sisa antrian.
2. Pengerjaan tugas akhir ini hanya pada analisis dan perancangan aplikasi ANTRI saja.
3. Pengerjaan tugas akhir ini tidak sampai pada tahap pembuatan aplikasi.

1.4 Tujuan Tugas Akhir

Tujuan yang diharapkan dari penelitian ini antara lain adalah sebagai berikut:

1. Menghasilkan dokumen analisis aplikasi ANTRI (SKPL) yang bisa menjadi rujukan dalam pengembangan sistem.

2. Menghasilkan dokumen perancangan aplikasi ANTRI (DDPL) yang bisa menjadi rujukan dalam pengembangan sistem.

1.5 Manfaat Tugas Akhir

Manfaat yang diharapkan dari tersusunnya tugas akhir ini adalah sebagai berikut:

1.5.1 Manfaat Akademis

Hasil penelitian ini diharapkan dapat menambah kajian ilmu pengetahuan dan dijadikan sebagai rujukan oleh peneliti selanjutnya dalam berkontribusi untuk membuat analisis dan perancangan aplikasi atau sistem di Perusahaan Pemerintahan yang bergerak di bidang jasa.

1.5.2 Manfaat bagi Perusahaan

RSIA Pondok Tjandra dapat mengetahui analisis dan perancangan sistem ANTRI, dan dapat dikembangkan dalam pembangunan aplikasi. Dengan adanya dokumen ini bisa menjadi pedoman oleh pengembang aplikasi untuk melakukan pembuatan dan bisa diterapkan pada Rumah Sakit, sehingga RSIA bisa memberikan pelayanan yang baik bagi pasien.

1.6 Relevansi

Tugas akhir ini disusun untuk memenuhi syarat kelulusan Sarjana. Peneliti mengangkat topik analisis dan perancangan aplikasi karena penelitian ini dapat menjadi acuan untuk melakukan pengembangan perangkat lunak diperusahaan. Keterkaitan penelitian ini dengan perkuliahan yang telah dipelajari oleh peneliti yakni pada mata kuliah Rancang Bangun Perangkat Lunak (RBPL) dan Interaksi Manusia dan Komputer(IMK). Penelitian ini termasuk dalam topik adopsi TI pada peta jalan penelitian salah satu laboratorium yang ada di Jurusan Sistem Informasi yaitu Manajemen Sistem Informasi (MSI), dengan sub topik analisis dan perancangan perangkat lunak.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan menjelaskan mengenai referensi teori terbaru, temuan dan bahan penelitian lain yang menjadi landasan informasi untuk memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini, dimana nantinya diharapkan akan menjadi kerangka berpikir untuk mengerjakan tugas akhir ini.

2.1 Studi Sebelumnya

Sebelum melakukan penelitian tugas akhir, dilakukan tinjauan pustaka terhadap tulisan dari beberapa penelitian sebelumnya yang sesuai dengan topik penelitian tugas akhir ini. Penelitian yang dijadikan acuan dalam pengerjaan tugas akhir ini disajikan pada tabel dibawah ini:

Tabel 2. 1 Studi Sebelumnya

Judul 1	Analisis dan Perancangan Aplikasi E NTRI
Penulis, Tahun	Nur Laili Sholichah dan teman-teman (kelompok 4 DDPL B), 2016
Metode	<i>Scrum</i>
Hasil yang didapatkan	Studi kasus yaitu analisis dan perancangan sistem antri pada rumah sakit umum, yang terintegrasi pada bagian apotek. Dari penelitian ini didapatkan bagaimana proses analisis dan perancangan sebuah aplikasi.
Keterkaitan Penelitian	Sebagai referensi untuk menentukan design apa saja yang dibutuhkan dalam perancangan aplikasi

Judul 2	Analisis Dan Perancangan Sistem Antrian Terintegrasi Untuk Peningkatan Keamanan, Produktivitas Dan Efisiensi Di Bank Indonesia
Penulis, Tahun	Renja Mantani, 2014
Metode	<i>Waterfall</i>
Hasil yang didapatkan	Analisa dilakukan pada aplikasi bank indonesia dengan menggunakan metode <i>waterfall</i> . Modul-modul pada sistem ini sangat kompleks.
Keterkaitan Penelitian	Sebagai referensi studi pustaka dalam mengerjakan penelitian topik tugas akhir ini karena dalam penelitian ini dijelaskan bagaimana analisis dan perancangan menggunakan metode <i>waterfall</i> .

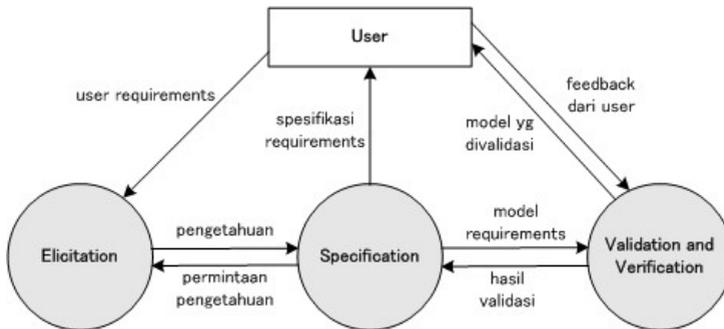
2.2 Dasar Teori

2.2.1 Requirement Engineering

Requirements Engineering adalah fase terdepan dari proses rekayasa perangkat lunak (*software engineering*), dimana *software requirements* dari *user* dan *customer* dikumpulkan, dipahami dan ditetapkan. Banyak definisi yang diungkapkan oleh para peneliti tentang *requirements engineering*. Satu definisi yang cukup jelas dan diterima secara umum adalah yang diuraikan oleh Pamela Zave [3]:

Requirements engineering adalah cabang dari *software engineering* yang mengurus masalah yang berhubungan dengan: tujuan (dunia nyata), fungsi, dan batasan-batasan pada sistem *software*. Termasuk hubungan faktor-faktor tersebut dalam menetapkan spesifikasi yang tepat dari suatu *software*, proses evolusinya baik berhubungan dengan masalah waktu maupun dengan *software* lain (dalam satu famili).

Hasil dari fase *requirements engineering* terdokumentasi dalam *requirements specification*. *Requirements specification* berisi kesepakatan bersama tentang permasalahan yang ingin dipecahkan antara pengembang dan customer, dan merupakan titik start menuju proses berikutnya yaitu *software design*. Sistemisasi proses negosiasi pengembang dan customer dalam *requirements engineering* dibagi dalam 3 proses besar yaitu: *elicitation*, *specification*, *validation and verification*. Formula ini kemudian juga dikenal dengan nama *The Three Dimensions of Requirements Engineering*. Proses *requirements engineering* ini dilakukan secara iterasi dengan mengakomodasi adanya *feedback* dari *customer (user)*. [4]



Gambar 2.1 Dimensi RE

2.2.1.1 Requirements Elicitation

Adalah proses mengumpulkan dan memahami *requirements* dari *user*. Kadang masalah yang muncul berakar dari *gap* masalah *knowledge domain* (perbedaan disiplin ilmu yang dimiliki). Customer adalah expert pada domain yang *softwarena* ingin dikembangkan (domain specialist), dilain pihak sang pengembang (*requirements analyst*) adakalanya sama sekali buta terhadap *knowledge domain* tersebut, meskipun tentu memahami dengan benar bagaimana sebuah *software* harus dikembangkan. *Gap knowledge domain* tersebut yang diharapkan bisa diatasi dengan adanya interaksi terus menerus dan berulang (iterasi) antara

pengembang dan *customer*. Proses interaksi tersebut kemudian dimodelkan menjadi beberapa teknik dan metodologi diantaranya adalah *interviewing*, *brainstorming*, *prototyping*, *use case*, dsb.

2.2.1.2 Requirements Specification

Setelah masalah berhasil dipahami, pengembang mendeskripsikannya dalam bentuk dokumen spesifikasi. Spesifikasi ini berisi tentang fitur dan fungsi yang diinginkan oleh *customer*, dan sama sekali tidak membahas bagaimana metode pengembangannya. IEEE mengeluarkan standard untuk dokumen spesifikasi *requirements* yang terkenal dengan nama IEEE *Recommended Practice for Software Requirements Specifications* [IEEE-830]. Dokumen spesifikasi *requirements* bisa berisi *functional requirements*, *performance requirements*, *external interface requirements*, *design constraints*, maupun *quality requirements*.

2.2.1.3 Requirements Validation and Verification

Setelah spesifikasi *requirements* berhasil dibuat, perlu dilakukan dua usaha: *Validation* (validasi), yaitu proses untuk memastikan bahwa *requirements* yang benar sudah ditulis. *Verification* (verifikasi), yaitu proses untuk memastikan bahwa *requirements* sudah ditulis dengan benar. Proses validasi dan verifikasi ini melibatkan *customer (user)* sebagai pihak yang menilai dan memberi *feedback* berhubungan dengan *requirements*.

2.2.2 Software Design

Software design atau desain sistem dapat didefinisikan sebagai: “Tahap setelah analisis dari siklus pengembangan sistem: pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi; menggambarkan bagaimana suatu sistem dibentuk.” [5]

Tahap desain sistem mempunyai dua maksud atau tujuan utama, yaitu:

1. Untuk memenuhi kebutuhan para pemakai sistem

2. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknik lainnya yang terlibat. Tujuannya lebih condong pada desain sistem yang terinci, yaitu pembuatan rancang bangun yang jelas dan lengkap untuk nantinya digunakan untuk pembuatan program komputernya.

Pengertian Desain Sistem Menurut Para Ahli adalah Menurut John Burch dan Garry Grudnitski dalam buku Analisa dan Desain, Sistem Informasi Pendekatan Terstruktur adalah : "Desain sistem dapat didefinisikan sebagai penggambaran dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi." Dari definisi diatas, maka dapat ditarik kesimpulan bahwa desain sistem adalah tahapan berupa penggambaran, perencanaan dan pembuatan dengan menyatukan beberapa elemen terpisah ke dalam satu kesatuan yang utuh untuk memperjelas bentuk sebuah sistem.

Menurut Yavri D. Mahyuzir dalam bukunya Pengolahan Data menyebutkan beberapa langkah yang perlu dilakukan pada proses desain sistem adalah : [5]

1. Menganalisa masalah dari pemakai (user), sasarannya adalah mendapatkan pengertian yang mendalam tentang kebutuhan-kebutuhan pemakai.
2. Studi kelayakan, membandingkan alternatif-alternatif pemecahan masalah untuk menentukan jalan keluar yang paling tepat.
3. Rancang sistem, membuat usulan pemecahan masalah secara logika.
4. Detail desain, melakukan desain sistem pemecahan masalah secara terperinci.
5. Penerapannya yaitu memindahkan logika program yang telah dibuat dalam bahasa yang dipilih, menguji program, menguji data dan outputnya.

6. Pemeliharaan dan evaluasi terhadap sistem yang telah diterapkan.

Tahap Analisis

1. Membuat struktur organisasi
2. Mendefinisikan kebutuhan informasi
3. Mendefinisikan kriteria kinerja system

Tahap Design

1. Menyiapkan rancangan
2. Membuat Context Diagram
3. Membuat ERD
4. Merancang Kamus Data
5. Membuat FlowChart
6. Merancang File (master, input, proses, temporary)
7. Merancang Dialog Input
8. Merancang Dialog Output
9. Menyiapkan konfigurasi system

2.2.3 Object Oriented Analysis and Design (OOAD)

Pendekatan sistem yang digunakan adalah menggunakan metode pendekatan yang berorientasi objek dengan menggunakan metode pemodelan *Unified Modeling Language*(UML). Melalui pendekatan berorientasi objek, OOAD adalah metode pengembangan sistem yang lebih menekankan objek dibandingkan dengan data atau proses. Ada beberapa ciri dari pendekatan ini, yaitu *object, inheritance, dan object class*. [6]

Object adalah struktur yang mengenkapsulasi atribut dan metode yang beroperasi berdasarkan atribut-atribut tadi. *Object* adalah abstraksi dari benda nyata dimana data dan proses diletakkan bersama untuk memodelkan struktur dan perilaku dari *object* dunia nyata.

Object class adalah sekumpulan objek yang berbagi struktur yang sama dan perilaku dari objek dunia nyata.

Inheritance merupakan properti yang muncul ketika tipe entitas atau object class disusun secara hierarki dan setiap tipe entitas atau object class menerima atau mewarisi atribut dan metode dari pendahulunya. [6]

2.2.3.1 Oriented Analysis (OOA)

Object Oriented Analysis adalah sebuah teknik yang model-driven yang mengintegrasikan data dan proses ke dalam konstruksi yang disebut objek. Model-model OOA adalah gambar-gambar yang mengilustrasikan objek-objek sistem dari berbagai macam perspektif seperti struktur, perilaku dan interaksi objek-objek. Contohnya adalah UML (*Unified Modeling Language*). [6]

2.2.3.2 Object Oriented Design (OOD)

Object Oriented Design adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

2.2.4 Unified Modeling Language (UML)

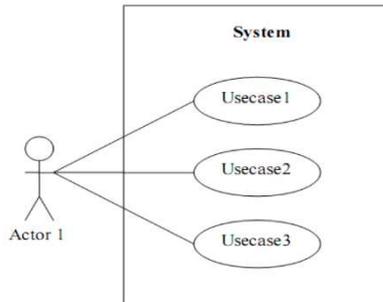
UML (*Unified Modeling Language*) adalah bahasa pemodelan, bukan metode. Bahasa pemodelan adalah metode mengekspresikan desain dari perangkat lunak yang telah dikembangkan. Tujuan dari pemodelan adalah ketika pengembang dan pengguna atau bisa antar anggota tim pengembang mendiskusikan tentang perangkat lunak yang dikembangkan. Pemodelan mempermudah dalam mendeskripsikan perangkat lunak yang dikembangkan. Pemodelan sebagai sarana analisis, visualisasi dan komunikasi dengan pihak terkait yang bermanfaat untuk mengamati dari perilaku dari perangkat lunak tersebut. [7]

UML memiliki beberapa diagram yang digunakan untuk menggambarkan suatu sistem. Tujuan pembuatan diagram ini adalah agar sistem mudah dimengerti oleh semua pihak, baik yang

teknis maupun nonteknis [7]. Beberapa contoh dari diagram tersebut, antara lain:

2.2.4.1 Use Case Diagram

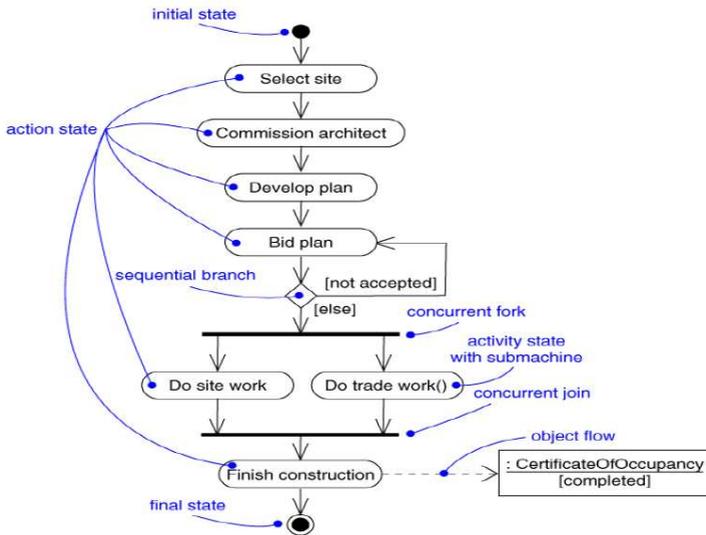
Menggambarkan sekumpulan use case dan actor dan hubungan antara mereka. Use case diagram mempunyai peranan penting dalam pengorganisasian dan pemodelan behavior dari sistem.



Gambar 2.2 Use Case Diagram

2.2.4.2 Activity Diagram

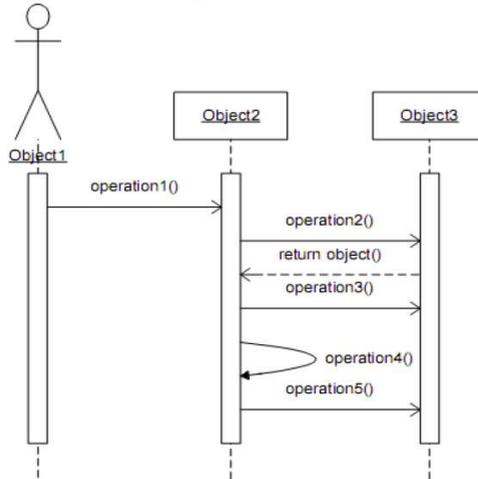
Menggambaran berbagai alir aktivitas dalam sistem yang sedang dirancang.



Gambar 2.3 Activity Diagram

2.2.4.3 Sequence Diagram

Merupakan diagram interaksi yang menekankan pada urutan waktu dari pertukaran *message*.



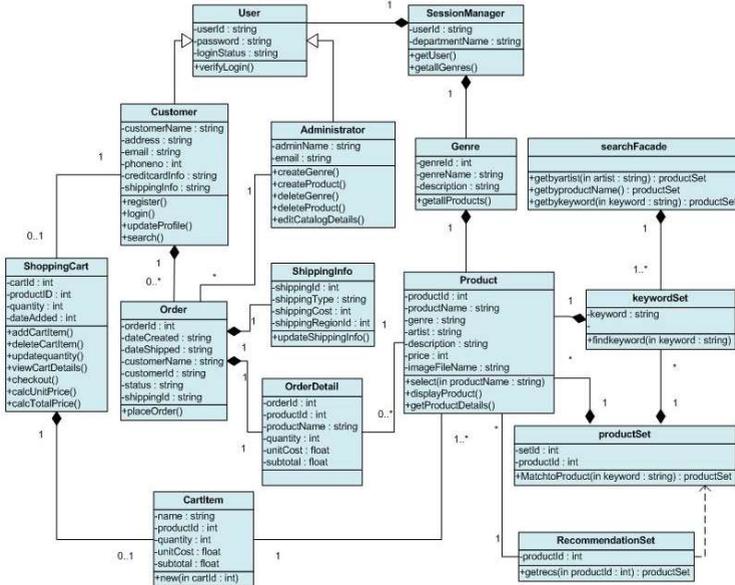
Gambar 2.4 Sequence Diagram

2.2.4.4 Class Diagram

Class diagram digunakan saat mengembangkan sebuah model sistem yang berorientasi objek untuk menunjukkan kelas-kelas yang ada dalam sistem dan asosiasi antara kelas-kelas tersebut. Sebuah kelas objek dapat dianggap sebagai definisi secara umum dari sebuah objek sistem. Sebuah asosiasi adalah penghubung antara kelas-kelas yang menunjukkan bahwa terdapat suatu hubungan antara kelas-kelas ini. Akibatnya, tiap kelas harus memiliki informasi dari kelas yang terkait.

Class diagram dalam UML dapat dinyatakan ada berbagai tingkat detail. Ketika kita mengembangkan sebuah model, mengidentifikasi objek penting, dan merespresentasikannya sebagai kelas-kelas. Cara termudah untuk menuliskannya adalah dengan menuliskan nama kelasnya dalam sebuah kotak atau

menuliskan asosiasi dengan menggambar sebuah garis diantara kelas-kelas. [8]



Gambar 2.5 Class Diagram

2.2.4.5 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah jaringan yang menggunakan data yang disimpan dari sistem secara abstrak. Tujuan dari ERD adalah untuk menunjukkan objek data dan relationship yang ada pada objek tersebut. Disamping itu ERD merupakan salah satu alat untuk perancangan dalam basis data. Simbol-simbol yang digunakan oleh ERD adalah sebagai berikut [9]:

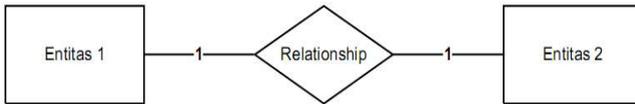
1. **Entity** : Objek dibedakan atau didefinisikan secara unik dengan objek lainnya, dimana semua informasi yang berkaitan. Simbol dari *entity* adalah kotak.
2. **Relationship** : Hubungan antara *entity* yang satu dengan *entity* yang lainnya. Simbol dari *relationship* adalah belah ketupat.

3. **Atribut** : penjelasan detail tentang *entity* atau *relationship*.
Simbol dari atribut adalah elips.

2.2.4.5.1 Kardinalitas

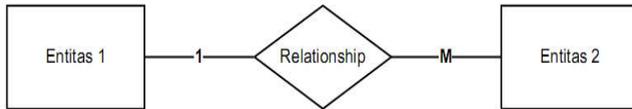
Kardinalitas bertujuan untuk menjelaskan batasan pada jumlah entity yang berhubungan melalui suatu relationship. Jenis-jenis kardinalitas adalah sebagai berikut:

1. One-to-one (1 : 1) : Perbandingan entity pertama dengan entity kedua satu berbanding satu.



Gambar 2.6 One-to-one relationship

2. One-to-many (1 : M) : Perbandingan antara entity pertama dengan entity kedua 1 berbanding banyak.



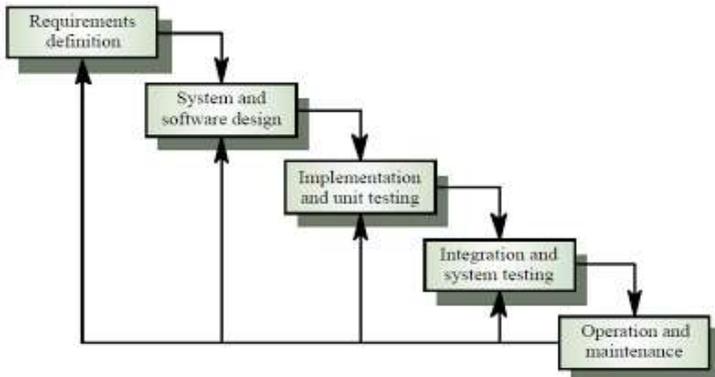
Gambar 2.7 One-to-many relationship

3. Many To One (M : 1) : Perbandingan entity pertama dengan entity kedua banyak berbanding satu.
Many To Many (M : M) : Entity pertama dengan entity kedua banyak berbanding banyak.

2.2.5 Waterfall Model

Waterfall model adalah metode dengan menganalogikan air terjun. Air yang melewati air terjun akan mengalir dari atas ke bawah, dan ketika air sudah sampai bawah, maka air tersebut sudah tidak bisa kembali ke atas, begitu juga dengan *waterfall model*. *Waterfall model* dilakukan secara bertahap atau beruntun, maka ketika terjadi kesalahan di tahap awal maka berakibat sangat fatal pada tahap berikutnya. Untuk melanjutkan ke tahap berikutnya, tahap sebelumnya harus diselesaikan terlebih dahulu secara sempurna.

Kekurangan *waterfall model* adalah menghabiskan waktu yang lebih lama. Kelebihan dari *waterfall model* adalah ketika



Gambar 2.8 *Waterfall model*

kebutuhan sudah diketahui dengan baik, maa dengan menggunakan waterfall model adalah keputusan yang tepat [8].

(Halaman ini sengaja dikosongkan)

BAB III METODOLOGI PENELITIAN

Bab ini akan berisikan alur atau tahapan metode penelitian yang dilakukan oleh penulis dalam penyelesaian tugas akhir ini agar penyelesaian tugas akhir ini mempunyai arah dan terstruktur. Metode yang digunakan adalah pengumpulan data, yaitu dengan cara observasi, wawancara, brainstorming dan studi pustaka. Dan metode pengembangan aplikasi yang dipakai adalah berorientasi objek dengan model pengembangan waterfall, tahapannya terdiri dari permulaan sistem, analisis sistem, perancangan sistem dan implementasi sistem. Namun dalam penelitian ini, hanya membahas tahap permulaan sistem sampai perancangan sistem. Dan juga pembuatan laporan sesudah metode waterfall model dilakukan. Laporan adalah dokumentasi berupa tugas akhir yang berisi hasil dari penelitian dan juga materi terkait penelitian. agar mudah dipahami, maka akan dijelaskan dalam metode pengerjaan tugas akhir dalam bentuk bagan seperti yang digambarkan pada Tabel 3.1.

Tabel 3.1 Metode Penelitian

Input	Proses	Output
<ul style="list-style-type: none"> • Buku software engineering • Jurnal software engineering dan sistem antrian • Paper software engineering dan sistem antrian • Internet 	1. Studi Literatur	<ul style="list-style-type: none"> • Pengetahuan tentang <i>software engineering</i> • Pengetahuan tentang metode <i>waterfall</i> • Pengetahuan tentang rekayasa perangkat lunak
<ul style="list-style-type: none"> • Dokumen SOP 	2. Analisis	Dokumen SKPL <ul style="list-style-type: none"> • Spesifikasi Kebutuhan

• Hasil Observasi, Brenchmark dan wawancara		<ul style="list-style-type: none"> • <i>Functional Requirement</i> • <i>Non-Functional Requirement</i> • <i>Usecase diagram</i>
Dokumen SKPL	3. Desain	<ol style="list-style-type: none"> 1. Sequence diagram 2. Class diagram 3. ERD 4. GUI
Skenario Testing	4. Testing	Validasi desain terhadap kebutuhan fungsional aplikasi
	5. Laporan	Laporan Final

3.1 Tahapan Persiapan

Tahapan persiapan adalah tahapan awal yang dilakukan untuk memulai tugas akhir ini. Tahapan persiapan terdiri dari studi literatur. Tahapan persiapan penting dilakukan sebagai pemahaman sebelum pengerjaan tugas akhir.

3.1.1 Studi Literatur

Tahapan persiapan bertujuan untuk pembelajaran dan pemahaman literatur yang berkaitan dengan permasalahan yang ada. Ada beberapa hal yang harus dipelajari seperti pembuatan dokumen analisis dan dokumen desain serta merancang *use-case* dengan baik dan benar. Studi literatur tidak terbatas hanya bersumber dari buku tentang *software engineering*, paper dan jurnal dengan ruang lingkup sistem antrian dan *software engineering* tetapi juga dari media lain seperti internet.

3.2 Tahapan Perancangan Perangkat Lunak

Tahapan perancangan perangkat lunak adalah tahapan yang paling penting pada pengerjaan tugas akhir ini. Pada tahapan perancangan perangkat lunak, terdapat 3 proses yang harus dilalui menurut metode *waterfall model*, yaitu analisis, desain dan juga testing.

3.2.1 Analisis

Pada proses analisis ini adalah tahap awal dalam tahapan perancangan perangkat lunak. Pada proses ini bertujuan untuk menganalisis kebutuhan dari perangkat lunak sistem antrian. Ada banyak yang perlu dianalisis didalam proses ini, yaitu analisis kebutuhan. Analisis proses bisnis, analisis kebutuhan pengguna dan analisis kebutuhan sistem. Masukkan proses analisis ini adalah dokumen SOP (*Standart Operational Procedure*) dan hasil dari observasi, brencmarking ke aplikasi yang serupa seperti E-ntri dan Antrian RSAB Soerya. Hasil dari proses analisis adah dokumen spesfikasi kebutuhan perangkat lunak.

3.2.1.1 Analisis Proses Bisnis

Analisis proses bisnis bertujuan untuk mengetahui proses bisnis yang saat ini ada di RSIA Pondok Tjandra dimana bermanfaat untuk perancangan perangkat lunak sistem antrian yang baru.

3.2.1.2 Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna bertujuan untuk mengetahui siapa saja pengguna dari aplikasi ANTRI dan mengidentifikasi kebutuhan apa saja dari masing-masing pengguna tersebut.

3.2.1.3 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem bertujuan untuk mengetahui kebutuhan dari perangkat lunak sistem antrian yang ingin dikembangkan. Analisis kebutuhan sistem terdiri dari *use-case*, spesifikasi fitur, dan kebutuhan fungsional dan non fungsional. Analisis kebutuhan

sistem nantinya digunakan untuk pengembangan desain perangkat lunak dari aplikasi antrian.

3.2.1.3.1 Use-case

Pembuatan *use-case* berpedoman dari fungsi-fungsi yang ada pada perangkat lunak yang akan dikembangkan. ada beberapa pengelompokan untuk membuat *use-case*, diantaranya adalah berdasarkan aktor dan fungsinya.

3.2.1.3.2 Spesifikasi Fitur

Spesifikasi fitur adalah pendefinisian fitur-fitur yang akan ada di perangkat lunak sistem antrian yang ingin dikembangkan. spesifikasi fitur ini disesuaikan dengan kebutuhan yang telah diidentifikasi.

3.2.1.3.3 Kebutuhan Fungsional dan Non-Fungsional

Kebutuhan fungsional adalah kebutuhan yang berhubungan dengan perangkat lunak yang ingin dibuat. Kebutuhan-kebutuhan tersebut nantinya dikelompokan berdasarkan fungsional perangkat lunak yang ingin dikembangkan. Kebutuhan non-fungsional adalah kebutuhan yang tidak secara langsung terkait dengan fitur dari perangkat lunak yang ingin dikembangkan. kebutuhan non fungsional sebagai batasan pada kebutuhan fungsional.

3.2.2 Desain

Pada proses desain ini bentuk dari diagram sistem antrian dan interface dari perangkat lunak sistem antrian yang baru akan terlihat. Proses desain ini menjadi penting karena tugas akhir ini tidak melakukan rancang bangun perangkat lunak, tetapi hanya mendesain sistem antrian yang baru dengan menggunakan *tools GUI Design Studio*. Untuk pembuatan diagram menggunakan *Microsoft Visio*, *Power Designer*, dan *Enterprise Architect*. Masukkan dari proses desain ini adalah dokumen spesifikasi perangkat lunak, sedangkan hasil dari proses desain ini adalah

validasi desain terhadap kebutuhan fungsional sistem. Pada proses desain ini dibagi menjadi 2, yaitu desain perilaku dan desain antarmuka.

3.2.2.1 Desain Perilaku

Desain perilaku bertujuan untuk pembuatan *use-case diagram*, *sequence diagram*, dan *class diagram* menggunakan UML.

3.2.2.2 Desain Antarmuka

Desain antarmuka adalah rancangan antarmuka dari perangkat lunak sistem antrian yang dikembangkan. desain antarmuka bertujuan untuk mengetahui bentuk dari perangkat lunak sistem antrian yang dikembangkan.

3.2.3 Validasi

Validasi bertujuan untuk memastikan bahwa system aplikasi yang dibangun sudah sesuai, yaitu produk yang memenuhi tujuan spesifik yang diharapkan. Peneliti menggunakan matrik keruntutan sebagai cara untuk memvalidasi desain aplikasi.

3.2.3.1 Matriks Keruntutan

Menjelaskan tentang keruntutan dari kesesuaian antara kebutuhan fungsional (KF), *Use-case (UC)*, *Use-case Scenario (UCS)*, *Sequence Diagram (SD)*, dan *Interface*. Dengan menggunakan matriks keruntutan, maka dapat diketahui apakah desain yang dibuat, telah sesuai dengan kebutuhan fungsional maupun *use-case* yang dibuat diawal.

3.3 Tahapan Penyelesaian

Tahapan penyelesaian adalah tahapan terakhir dari tugas akhir ini. Tahapan ini akan menghasilkan laporan berupa buku tugas akhir. Tahapan penyelesaian juga bertujuan untuk dokumentasi dari proses tugas akhir dan penelitian yang sudah dilakukan.

3.3.1 Laporan

Proses pembuatan laporan adalah proses akhir yang dilakukan dari proses pembuatan tugas akhir ini. Hasil dari proses pembuatan laporan adalah buku tugas akhir yang berjudul “ANALISIS DAN PERANCANGAN APLIKASI “ANTRI” BERBASIS WEBSITE DAN MOBILE ANDROID.

BAB IV PERANCANGAN

Bab ini menjelaskan perancangan penelitian tugas akhir yang terkait dengan rancangan penelitian, subyek dan obyek penelitian, serta strategi pelaksanaan yang dilakukan mulai dari awal penelitian hingga hasil akhir dari tugas akhir ini.

4.1 Studi Literatur

Studi literatur yang dilakukan seperti pendalaman proses bisnis RSIA Pondok TJandra yang berhubungan dengan rumusan masalah dalam tugas akhir ini. Selain itu, dilakukan studi pustaka mengenai materi yang berhubungan dengan Rancang Bangun Perangkat Lunak, metode *waterfall* dan metode *OOAD*. Referensi didapatkan dari beberapa laman *website*, *blog* yang berdasarkan jurnal resmi, dokumen *offline* dari perusahaan, jurnal ilmiah, artikel populer, buku, dan tugas akhir.

4.2 Perancangan Perangkat Lunak

Tahapan perancangan perangkat lunak adalah tahapan yang paling penting pada pengerjaan tugas akhir ini. Pada tahapan perancangan perangkat lunak, terdapat 3 proses yang harus dilalui menurut metode *waterfall model*, yaitu analisis, desain dan juga *testing*.

4.2.1 Analisis

Pada tahap ini bertujuan untuk menganalisis kebutuhan dari perangkat lunak sistem antrian. Ada banyak yang perlu dianalisis didalam proses ini, yaitu analisis kebutuhan. Analisis proses bisnis, analisis kebutuhan pengguna dan analisis kebutuhan sistem. Masukkan proses analisis ini adalah dokumen SOP (*Standart Operational Procedure*), wawancara, observasi, dan brenchmarking ke aplikasi yang serupa seperti E-ntri dan Antrian RSAB Soerya. Hasil dari proses analisis adalah dokumen spesifikasi kebutuhan perangkat lunak.

Wawancara

Wawancara dilakukan pada narasumber yaitu 1 orang staf administrasi umum dan 2 orang Pasien. Tujuan dari wawancara ini dilakukan untuk mendapatkan informasi seperti:

1. Kondisi saat ini dari proses bisnis dan peran Teknologi Informasi (TI) dalam RS,
2. Permasalahan yang sering terjadi.

Review Dokumen

Review dokumen pada penelitian ini dilakukan untuk mengetahui struktur organisasi, proses bisnis, dokumen terkait di RSIA Pondok Tjandra yang didapatkan dari perusahaan secara *offline*.

Observasi

Observasi dilakukan untuk mengetahui secara langsung bagaimana kondisi *existing* mengenai bisnis dan dukungan teknologi informasi (TI) yang ada di RSIA Pondok Tjandra.

4.2.1.1 Analisis Proses Bisnis

Analisis proses bisnis bertujuan untuk mengetahui proses bisnis yang saat ini ada di RSIA Pondok Tjandra dimana bermanfaat untuk perancangan perangkat lunak sistem antrian yang baru.

4.2.1.1.1 Gambaran Sistem Yang Berjalan

Pada Bagian ini akan dijelaskan bagaimana gambaran sistem yang sedang berjalan di RSIA Pondok Tjandra. Gambaran sistem ini didapatkan dari pengamatan dan hasil wawancara yang dilakukan terhadap pihak Rumah Sakit. Bab ini menjelaskan bagaimana alur-alur proses dari pendaftaran pemeriksaan, proses pemeriksaan sampai selesai, hingga pasien menuju apotek dan pasien pulang. Gambaran sistem ini akan dibuat bentuk FlowChart.

4.2.1.1.2 Gambaran Sistem Yang Baru

Pada Bagian ini akan dijelaskan bagaimana gambaran sistem yang akan dibuat di RSIA Pondok Tjandra. Gambaran sistem ini dirancang berdasarkan analisis pada permasalahan yang didapat. Gambaran sistem ini akan dibuat bentuk FlowChart.

4.2.1.2 Analisis Kebutuhan Pengguna

4.2.1.2.1 Pengguna dan Tanggung jawab

Analisis kebutuhan bertujuan untuk mengetahui siapa saja pengguna dari aplikasi ANTRI dan mengidentifikasi kebutuhan apa saja dari masing-masing pengguna tersebut. Pada bagian ini dijelaskan siapa saja pengguna dari aplikasi “ANTRI” dan apa kebijakan-kebijakan pada masing-masing penggunaannya.

4.2.1.3 Analisis Kebutuhan Sistem

4.2.1.3.1 Aturan Sistem Antrian

Pada Bagian ini dijelaskan aturan-aturan yang ada pada aplikasi “ANTRI” ini. Bagaimana aturan untuk masing-masing para penggunaannya dan bagaimana wewenang yang berlaku bagi penggunaannya. Analisis kebutuhan sistem bertujuan untuk mengetahui kebutuhan dari perangkat lunak sistem antrian yang ingin dikembangkan. Analisis kebutuhan sistem terdiri dari *use-case*, spesifikasi fitur, dan kebutuhan fungsional dan non fungsional. Analisis kebutuhan sistem nantinya digunakan untuk pengembangan desain perangkat lunak dari aplikasi antrian.

4.2.1.3.1.1 Use-case

Pembuatan *use-case* berpedoman dari fungsi-fungsi yang ada pada perangkat lunak yang akan dikembangkan. ada beberapa pengelompokkan untuk membuat *use-case*, diantaranya adalah berdasarkan aktor dan fungsinya.

Table 2 Contoh Tabel Use-Case

	Kode use-case	Use-case	Aktor
KF-XX	UC-XX
	UC-XX

4.2.1.3.1.2 Spesifikasi Fitur

Spesifikasi fitur adalah pendefinisian fitur-fitur yang akan ada di perangkat lunak sistem antrian yang ingin dikembangkan. spesifikasi fitur ini disesuaikan dengan kebutuhan yang telah diidentifikasi.

4.2.1.3.1.3 Kebutuhan Fungsional dan Non-Fungsional

Kebutuhan fungsional adalah kebutuhan yang berhubungan dengan perangkat lunak yang ingin dibuat. Kebutuhan-kebutuhan tersebut nantinya dikelompokkan berdasarkan fungsional perangkat lunak yang ingin dikembangkan. Kebutuhan non-fungsional adalah kebutuhan yang tidak secara langsung terkait dengan fitur dari perangkat lunak yang ingin dikembangkan. kebutuhan non fungsional sebagai batasan pada kebutuhan fungsional.

Table 3 Contoh Tabel Kebutuhan Fungsional dan Non-Fungsional

KF-XX
KF-XX
KNF-XX
KNF-XX

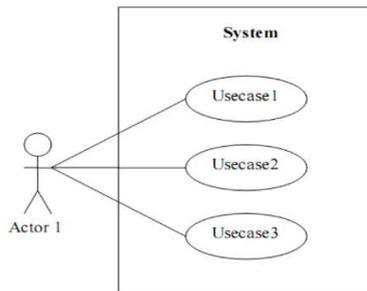
4.2.2 Desain

Pada proses desain ini bentuk dari diagram sistem antrian dan interface dari perangkat lunak sistem antrian yang baru akan terlihat. Proses desain ini menjadi penting karena tugas akhir ini tidak melakukan rancang bangun perangkat lunak, tetapi

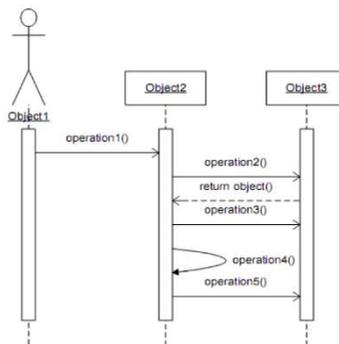
hanya mendesain sistem antrian yang baru dengan menggunakan *tools GUI Design Studio*. Untuk pembuatan diagram menggunakan *Microsoft Visio*, *Power Designer*, dan *Enterprise Architect*. Masukkan dari proses desain ini adalah dokumen spesifikasi perangkat lunak, sedangkan hasil dari proses desain ini adalah validasi desain terhadap kebutuhan fungsional sistem. Pada proses desain ini dibagi menjadi 2, yaitu desain perilaku dan desain antarmuka.

4.2.2.1 Desain Perilaku

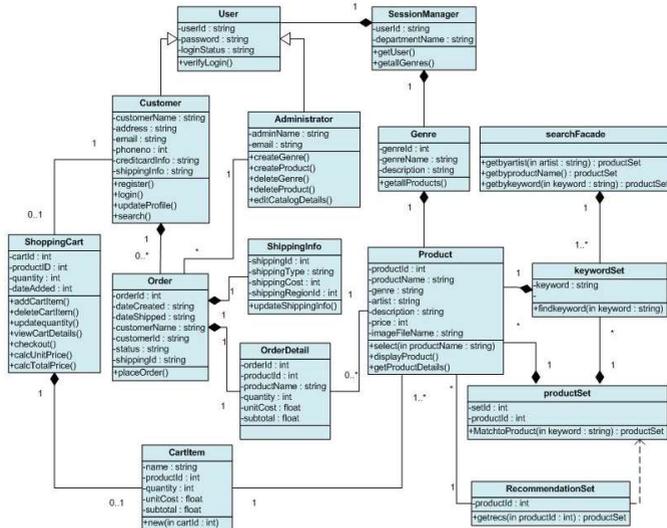
Desain perilaku bertujuan untuk pembuatan *use-case diagram*, *sequence diagram*, dan *class diagram* menggunakan UML.



Gambar 4.1 Contoh Use-case Diagram



Gambar 9 Contoh Sequence Diagram



Gambar 103 Contoh Class Diagram

4.2.2.2 Desain Antarmuka

Desain antarmuka a

adalah rancangan antarmuka dari perangkat lunak sistem antrian yang dikembangkan. desain antarmuka bertujuan untuk mengetahui bentuk dari perangkat lunak sistem antrian yang dikembangkan.

The image shows a web application window titled "ANTRI". At the top right, it says "Selamat Datang, 'Admin'". Below this, there are two tabs: "Beranda" and "Profil". The "Profil" tab is active, showing a form with the following fields: "Nama" (with a text input box), "Alamat" (with a text input box), "password" (with a text input box containing "pwd"), and "tulis kembali pwd" (with a text input box containing "pwd 2"). At the bottom of the form, there are two buttons: "simpan" and "batal".

Gambar 11.4 Contoh Desain Antarmuka (GUI)

4.2.3 Validasi

Validasi bertujuan untuk memastikan bahwa system aplikasi yang dibangun sudah sesuai, yaitu produk yang memenuhi tujuan spesifik yang diharapkan. Peneliti menggunakan matriks keruntutan sebagai cara untuk memvalidasi desain aplikasi.

4.2.3.1 Matriks Keruntutan

Menjelaskan tentang keruntutan dari kesesuaian antara kebutuhan fungsional (KF), *Use-case (UC)*, *Use-case Scenario (UCS)*, *Sequence Diagram (SD)*, dan *Interface*. Dengan menggunakan matriks keruntutan, maka dapat diketahui apakah desain yang dibuat, telah sesuai dengan kebutuhan fungsional maupun *use-case* yang dbuat diawal.

Tabel 4.1 Contoh Matriks Keruntutan

Kebutuhan Fungsional	Kode use-case	Use-case	Sequence Diagram	Interface Admin	Interface AssDok	Interface Pasien
KF-xx	UC-xx
	UC-xx

4.3 Tahapan Penyelesaian

Tahapan penyelesaian adalah tahapan terakhir dari tugas akhir ini. Tahapan ini akan menghasilkan laporan berupa buku tugas akhir. Tahapan penyelesaian juga bertujuan untuk dokumentasi dari proses tugas akhir dan penelitian yang sudah dilakukan. Proses pembuatan laporan adalah proses akhir yang dilakukan dari proses pembuatan tugas akhir ini. Hasil dari proses pembuatan laporan adalah buku tugas akhir yang berjudul “ANALISIS DAN PERANCANGAN APLIKASI “ANTRI” BERBASIS WEBSITE DAN MOBILE ANDROID

BAB V IMPLEMENTASI

Bab ini akan menjelaskan bagaimana proses implementasi dalam pengerjaan Tugas Akhir ini yang sesuai dengan perancangan yang telah disusun pada bab sebelumnya.

5.1 Proses Pelaksanaan Penelitian

5.1.1 Pengambilan Data

Pengambilan data ditujukan untuk mengumpulkan segala data yang terkait dengan implementasi aplikasi “ANTRI” serta bagaimana proses bisnis dan proses Teknologi Informasi (TI) pada RSIA Pondok Tjandra. Pengambilan data dibagi menjadi tiga cara, yaitu dengan cara wawancara, brencmark ke beberapa aplikasi serupa dan observasi di lingkungan rumah sakit.

5.1.1.1 Jadwal Pengambilan Data

Jadwal pengumpulan data adalah waktu pelaksanaan yang digunakan untuk mengumpulkan data terkait analisis proses bisnis dan permasalahan yang ada pada RSIA Pondok Tjandra, Jadwal pengumpulan data tersebut tercantum pada tabel dibawah ini:

Tabel 5. 1 Jadwal Pengambilan Data

Hari/Tanggal	Narasumber	Data
Senin, 17 April 2017	Imatus Solichah (Staf Administrasi Umum)	<ul style="list-style-type: none"> • Dokumen SOP RS • Proses Bisnis Perusahaan • Permasalahan yang terjadi
Senin, 24 April 2017	Anita(Pasien) Nanik(Pasien)	Permasalahan yang terjadi pada pasien
Senin, 01 Mei 2017	Imatus Solichah (Staf Administrasi Umum)	Wawancara tentang pengguna aplikasi

Hari/Tanggal	Narasumber	Data
Kamis, 01 Juni 2017	Imatus Solichah (Staf Administrasi Umum)	Wawancara tentang Desain Aplikasi

5.2 Hambatan dan Rintangan dalam Pelaksanaan

Dalam pelaksanaan implementasi aplikasi “ANTRI” menemui beberapa hambatan, tetapi tidak banyak pengaruhnya terhadap pengerjaan analisis dan perancangan aplikasi ini. Di antaranya adalah :

1. Petugas RS yang sibuk karena bekerja shift dan setiap shift hanya terdapat satu prang administrasi saja.
2. Hanya beberapa dokumen saja yang di perlihatkan kepada penulis.

5.3 Hasil Review Dokumen

Dokumen yang digunakan untuk mendukung penyelesaian tugas akhir ini didapatkan dari staf administrasi RSIA Pondok Tjandra. Dokumen-dokumen tersebut berisi mengenai informasi-informasi Rumah Sakit dan *Annual Report*.

5.3.1 Rumah Sakit Ibu dan Anak (RSIA) Pondok Tjandra

Rumah Sakit Ibu dan Anak (RSIA) Pondok Tjandra adalah rumah sakit milik swasta/lainnya Sidoarjo yang berbentuk RS Bersalin, yang dinaungi oleh Yayasan Baskara Husada, organisasi sosial dan termuat kedalam Rumah Skait Kelas Non-K. RS ini mendapat izin dari kepala Dinas Kesehatan Kabupaten Sidoarjo dengan nomor surat 551.41/34/404.3.2/2012. Saat ini RS ini dipimpin oleh Dr. drg. Sudiby, M. Kes. RS ini bertempat di Jl. Mangga I E nomor 225, Pondok Tjandra Indah 61256.

5.3.2 Visi dan Misi

Visi

1. Memberikan pelayanan kesehatan ibu dan Anak yang bermutu dengan kualitas standar ilmiah, bersih, ramah, jujur, tepat waktu dan terjangkau oleh masarakat.
2. Selalu memelihara citra dan ikatan kekeluargaan dengan pasien.
3. Berorientasi pada peningkatan kesehatan, kesejahteraan ibu dan anak serta wanita dari usia mudan sampai usia lanjut.

Misi

1. Menjadikan pasien sebagai pusat perhatian pelayanan.
2. Dikelola secara efisien dan profesional dilakukan oleh tenaga-tenaga terdidik, terlatih dan berkemampuan tinggi dalam memberikan pelayanan.
3. Berperan serta dalam membentuk tenaga pelaksana berakhlaq mulia, kreatif, santun dan beriman.
4. Berperan serta dalam menunjang pembangunan sistem kesehatan nasional, khususnya ibu dan anak serta wanita usia muda sampai usia lanjut agar tetap sehat dan sejahtera.

5.3.3 Tujuan

1. Memberikan pelayanan prima dan profesional berdasarkan standar prosedur operasional yang ditetapkan
 - a. Menyelenggarakan pelayanan yang bermutu, memuaskan dan profesional berdasarkan standar prosedur operasional yang ditetapkan
 - b. Senantiasa mengikuti perkembangan ilmu pengetahuan dan teknologi kedokteran dan kesehatan mutakhir
 - c. Mengembangkan kompetensi dasar dan terapan untuk meningkatkan mutu pelayanan
 - d. Menggalang dan mengembangkan kemitraan dengan berbagai pihak untuk menjalin jaringan kerjasama yang saling menguntungkan

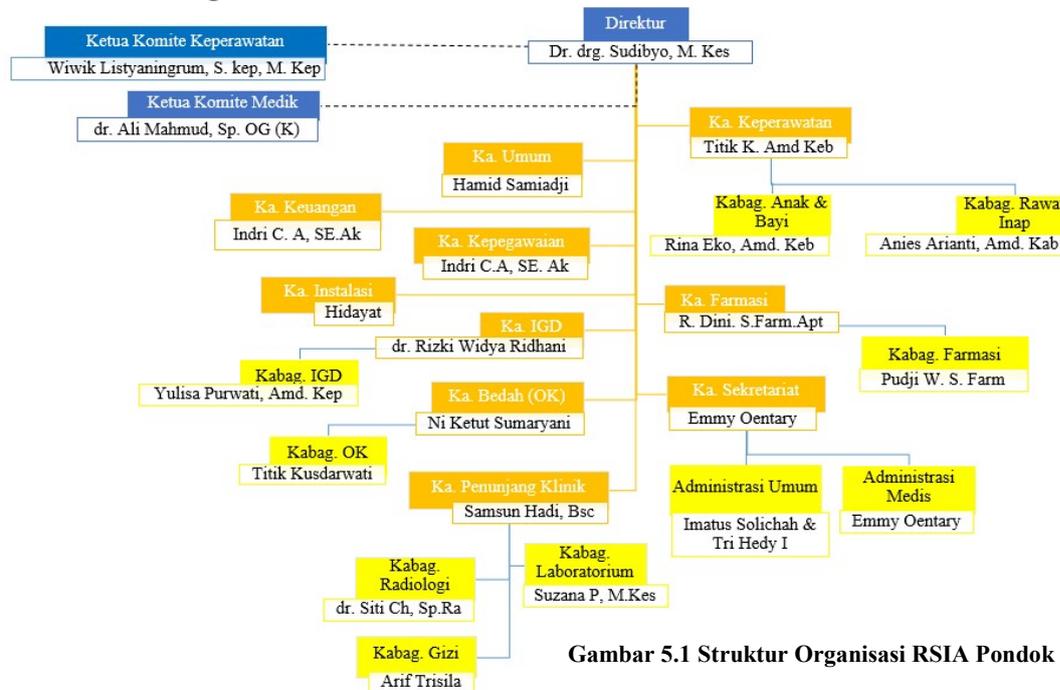
- e. Mewujudkan tingkat kepuasan konsumen internal (petugas RS), maupun konsumen eksternal (pasien) secara optimal
 - f. Memberdayakan seluruh potensi sumber daya dan dimiliki rumah sakit.
2. Menjadi rumah sakit ibu dan anak yang mampu mewujudkan fungsinya sebagai pelayanan kesehatan ibu dan anak, serta mampu memberikan manfaat untuk masyarakat.

5.3.4 Fasilitas Pelayanan

Sarana dan Prasarana relatif cukup memadai dalam memenuhi kebutuhan pelayanan kesehatan kesehatan ibu dan anak, antara lain:

1. Pelayanan dokter spesialis Obstetri dan Ginekologi
2. Pelayanan dokter spesialis anak
3. Pelayanan dokter spesialis anestesi
4. Pelayanan dokter umum
5. Pelayanan instalasi gawat darurat
6. Pelayanan instalasi farmasi
7. Pelayanan laboratorium
8. Pelayanan radiologi
9. Pelayanan Ultrasonografi (USG) 4 Dimensi
10. Pelayanan Onkologi
11. Pelayanan Konsultasi Gizi
12. Pelayanan Ambulans

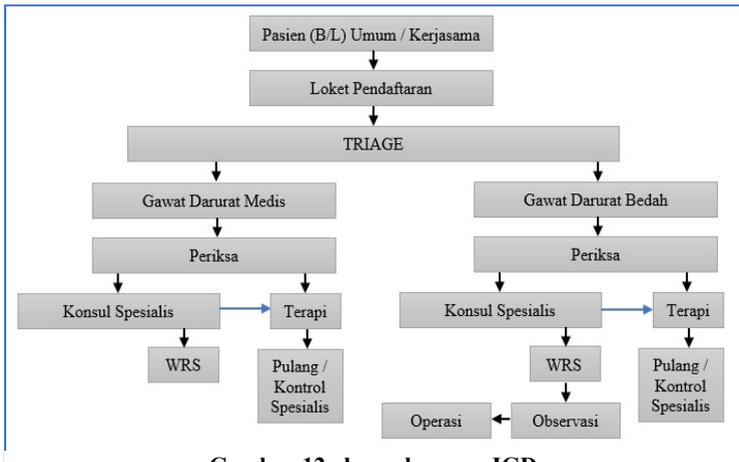
5.3.5 Struktur Organisasi



Gambar 5.1 Struktur Organisasi RSIA Pondok Tjandra

5.3.6 Alur Pelayanan IGD

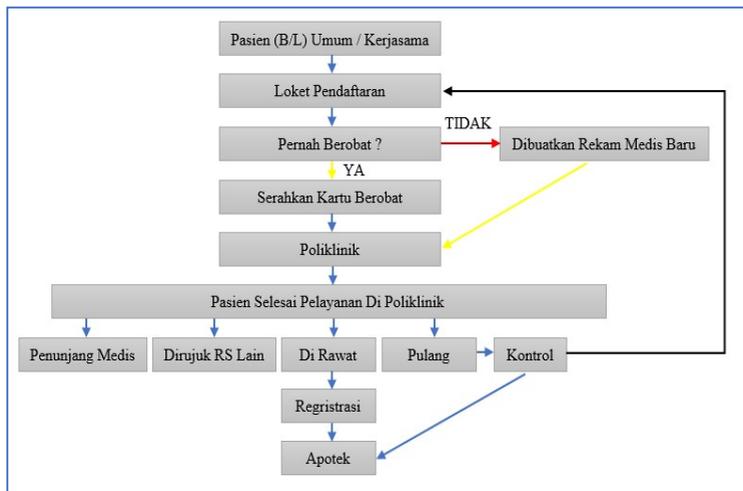
Pada gambar dibawah ini dijelaskan alur pelayan ketika ada pasien datang ke igd. Pasien umum maupun kerjasama harus daftar terlebih dahulu. Kemudian dokter atau perawat yang jaga menentukan beratnya cedera atau penyakit untuk menentukan tindakan perawatan seperti apa. Jika kondisi pasien tidak seberapa parah atau hanya membutuhkan perawatan medis, maka dokter langsung memeriksa pasien, kemudian dari hasil pemeriksaan akan diputuskan apakah perlu konsultasi ke dokter spesialis atau terapi, setelah keadaan pasien membaik bisa pulang. Jika kondisi pasien gawat darurat bedah maka akan dilakukan pemeriksaan dan observasi. Dan jika sudah sehat maka bisa langsung pulang. Apabila dilakukan penanganan maka pasien harus rawat inap.



Gambar 12 alur pelayanan IGD

5.3.7 Alur Pelayanan Rawat Jalan

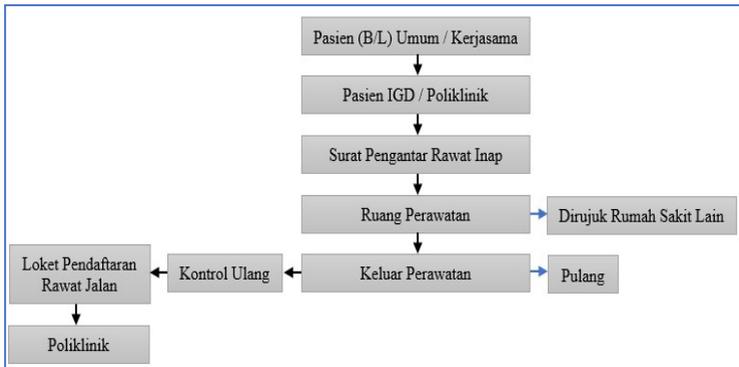
Pada gambar dibawah ini dijelaskan alur pelayanan rawat jalan pada RSIA Pondok Tjandra. Pasien umum maupun kerjasama harus daftar terlebih dahulu di loket pendaftaran. Jika pasien sudah pernah berobat, pasien hanya menyerahkan kartu berobat saja, dan saat ini bisa mendaftar melalui telepon untuk mendapatkan antrian. Jika pasien baru makan harus mengisi formulir data diri untuk dibuatkan rekam medis baru. Kemudian masuk ke poli untuk pemeriksaan. Pasien selesai pelayanan di poliklinik, pasien membayar di loket yg sama dengan pendaftaran tadi. Jika pasien memerlukan penunjang medis lainnya bisa langsung ke laboratorium atau ruang khusus lainnya. Bisa juga pasien di rujuk ke RS lain. Di rawat jika dibutuhkan perawatan. Dan bisa pulang lalu kontrol di waktu yang sudah ditentukan dokter. Dan ke apotek untuk menebus resep/membeli obat.



Gambar 13 alur pelayanan rawat jalan

5.3.8 Alur Pelayanan Rawat Inap

Pada gambar dibawah ini dijelaskan tentang pelayanan rawat inap. Pasien yang akan rawat inap disini bisa dari pasien IGD atau poliklinik, dan bisa juga dari rujukan RS lain. Menunjukkan surat pengantar rawat inap, kemudian memilih ruang perawatan kelas berapa, jika dalam masa perawatan pasien tidak ada perubahan atau mengalami hal-hal yang tidak bisa di tangani di RS ini, maka pasien di rujuk ke RS lain yang lebih lengkap peralatan medisnya. Setelah perawatan, jika dokter sudah menyatakan pasien sembuh, maka pasien diperbolehkan pulang. Dan di sarankan kontrol ulang dalam waktu yang sudah ditentukan oleh dokter. Ketika kontrol pasien juga mendaftar melalui loket pendaftaran poli.



Gambar 14 Alur pelayanan rawat inap

BAB VI HASIL DAN PEMBAHASAN

Bab ini akan menjelaskan mengenai hasil dan pembahasan secara keseluruhan dari penelitian ini terkait analisis dan perancangan aplikasi “ANTRI” pada RSIA Pondok Tjandra.

6.1 Perancangan Perangkat Lunak

Tahapan perancangan perangkat lunak adalah tahapan yang paling penting pada pengerjaan tugas akhir ini. Pada tahapan perancangan perangkat lunak, terdapat 3 proses yang harus dilalui menurut metode *waterfall model*, yaitu analisis, desain dan juga *testing*.

6.1.1 Analisis

Pada tahap ini bertujuan untuk menganalisis kebutuhan dari perangkat lunak sistem antrian. Ada banyak yang perlu dianalisis didalam proses ini, yaitu analisis kebutuhan. Analisis proses bisnis, analisis kebutuhan pengguna dan analisis kebutuhan sistem. Masukkan proses analisis ini adalah dokumen SOP (*Standart Operational Procedure*), wawancara, observasi, dan benchmarking ke aplikasi yang serupa seperti E-ntri dan Antrian RSAB Soerya. Hasil dari proses analisis adalah dokumen spesifikasi kebutuhan perangkat lunak.

Wawancara

Wawancara dilakukan pada narasumber yaitu 1 orang staf administrasi umum dan 2 orang Pasien. Tujuan dari wawancara ini dilakukan untuk mendapatkan informasi seperti:

1. Kondisi saat ini dari proses bisnis dan peran Teknologi Informasi (TI) dalam RS,
2. Permasalahan yang sering terjadi.

Review Dokumen

Review dokumen pada penelitian ini dilakukan untuk mengetahui struktur organisasi, proses bisnis, dokumen terkait

di RSIA Pondok Tjandra yang didapatkan dari perusahaan secara *offline*.

Observasi

Observasi dilakukan untuk mengetahui secara langsung bagaimana kondisi *existing* mengenai bisnis dan dukungan teknologi informasi (TI) yang ada di RSIA Pondok Tjandra.

6.1.1.1 Analisis Proses Bisnis

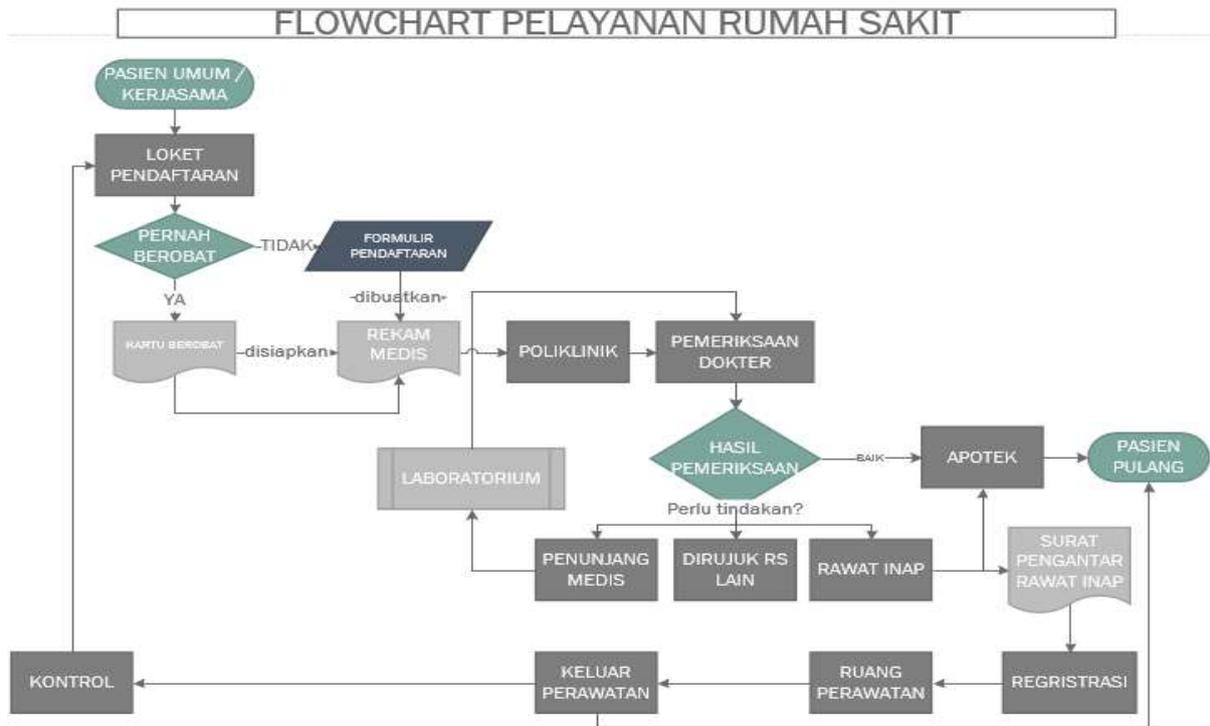
Analisis proses bisnis bertujuan untuk mengetahui proses bisnis yang saat ini ada di RSIA Pondok Tjandra dimana bermanfaat untuk perancangan perangkat lunak sistem antrian yang baru.

6.1.1.1.1 Gambaran Sistem Yang Berjalan

Berdasarkan pengamatan yang penulis lakukan melalui wawancara dengan pihak rumah sakit dan observasi, maka penulis menyimpulkan proses bisnis yang berjalan pada RSIA Pondok Tjandra diuraikan sebagai berikut :

Pasien yang ingin mendaftarkan diri, menghadap ke bagian administrasi, pihak bagian pendaftaran akan memberikan formulir. Setelah formulir di isi oleh pasien maka pasien akan diberikan kartu berobat. Sedangkan bagi pasien yang sudah terdaftar dan telah memiliki kartu berobat tinggal mendaftarkan diri di bagian administrasi dan akan diberikan nomor antrian pasien. Pasien yang sudah terdaftar di RSIA bisa mendaftarkan diri melalui telepon dengan menyebutkan nomer registrasi dan akan diberitahukan nomor antrian pasien, ketika pasien telah sampai di rumah sakit pasien tersebut tinggal menkonfirmasi kedatangannya dan menunggu sesuai dengan nomor antrian. Pasien yang memiliki asuransi kesehatan yang bekerjasama dengan pihak rumah sakit, dapat memberikan kartunya pada saat mendaftar maka pihak administrasi akan memberikan surat pengantar asuransi. Surat tersebut akan digunakan pada saat proses pembayaran setelah pasien selesai. Pasien akan dipanggil sesuai antrian, dokter akan memeriksa pasien dan memberitahukan keadaan pasien. Dokter akan menuliskan hasil

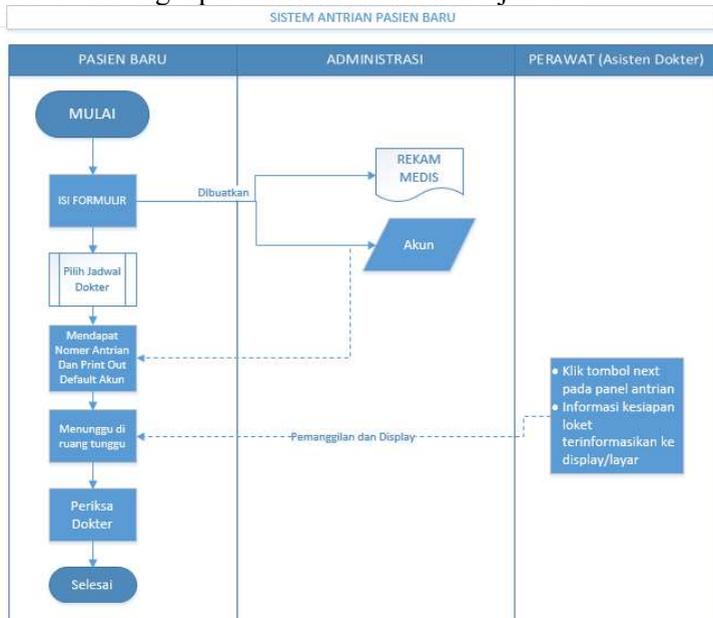
diagnosanya pada buku pasien. Dokter juga akan memberikan salinan resep pada pasien jika diperlukan. Setelah pasien diperiksa maka pasien melakukan pembayaran dibagian administrasi sesuai dengan pemeriksaan dokter. Buku pasien disimpan oleh RSIA, ketika pasien akan berobat lagi maka pasien hanya melakukan pendaftaran dan mengkonfirmasi nomer registasi pada bagian administrasi, kemudian petugas administrasi mencari dan menyiapkan buku pasien tersebut. Bagi pasien yang akan dirawat harus mengisi formulir persetujuan untuk dirawat. Jika dilakukan tindakan operasi pada saat perawatan maka pasien harus mengisi dan menanda tangani formulir persetujuan tindakan. Setiap pasien yang telah selesai masa perawatan dan diizinkan pulang/keluar rumah sakit, sebelum pulang harus melunasi terlebih dahulu semua biaya perawatan. Untuk mengetahui rincian biaya selama perawatan pasien menemui bagian administrasi. Pihak rumah sakit tidak menerima keringanan / discount biaya perawatan pasien. Proses pembayaran juga dilakukan pada bagian administrasi, pasien akan membayar sesuai dengan penanganan dan pemeriksaan dokter. Pada pasien rawat jalan pasien harus melunasi pembayaran pada saat itu juga. Bagi pasien yang memiliki kartu asuransi maka akan mendapatkan keringanan pembayaran dan di data oleh pihak administrasi sesuai dengan data asuransi pasien. Rincian pembayaran akan diberikan oleh pihak administrasi. Pembayaran dapat dilakukan secara tunai dan non tunai.



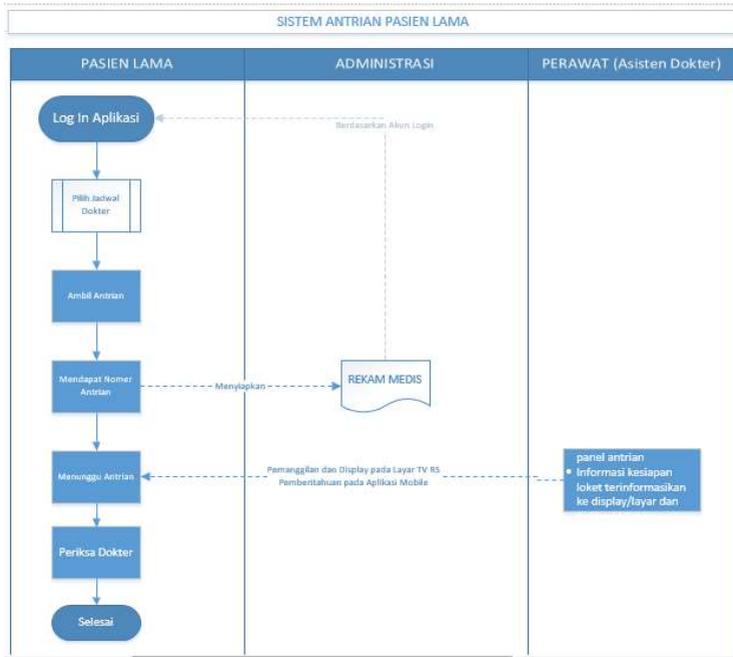
Gambar 6.15 Flowchart Pelayanan RS

6.1.1.1.2 Gambaran Sistem Yang Baru

Berikut adalah flowchart sistem yang baru untuk proses antrian pasien baru dan pasien lama. User pada aplikasi ini tidak perlu melakukan registrasi pada aplikasi, melainkan secara otomatis mendapatkan user dan password default apabila pasien sudah terdaftar sebagai pasien di RSIA Pondok Tjandra.



Gambar 16 Flowchart sistem baru untuk pasien baru



Gambar 17 Flowchart sistem baru untuk pasien lama

6.1.1.2 Analisis Kebutuhan Pengguna

6.1.1.2.1 Pengguna dan Tanggung jawab

Berdasarkan hasil dari penggalian kebutuhan, terdapat tiga(3) pengguna dari sistem antrian yang baru, yaitu:

1) *Administrator*

Administrator bertugas untuk mengelola akun user, seperti menghapus, menambahkan, mengubah, menonaktifkan dan mengaktifkan kembali akun user. *Administrator* bertugas untuk mengelola jadwal praktek dokter, seperti menambah default jadwal praktek, menambah jadwal praktek *urgent*, menonaktifkan jadwal praktek

2) *Pasien*

Pasien adalah orang-orang yang menggunakan aplikasi antrian untuk bisa bertransaksi atau mengambil nomer antrian pemeriksaan dokter.

3) Asisten Dokter(Perawat)

Asisten Dokter bertugas untuk menyetting antrian, jika ada keadaan darurat harus membuka antrian dengan jadwal baru atau menutup sebelum waktunya selesai. Asisten dokter juga bisa melihat berapa banyak pasien yang sedang mengantri. Dan juga asisten dokter bertugas menekan tombol **Next** untuk melanjutkan antrian selanjutnya jika pasien yang didalam ruang periksa sudah selesai.

6.1.1.2.2 Kebijakan Pengguna

Kebijakan-kebijakan yang berkaitan dengan pengguna dari sistem antrian yang baru dibagi menurut pengguna.

1) *Administrator*

Kebijakan-kebijakan dari administrator adalah sebagai berikut:

- Administrator dilarang menyebarkan data pengguna sistem, antrian dan juga data pasien.
- Administrator dilarang menghapus, menonaktifkan dan mengaktifkan kembali user sebelum ada surat atau perintah dari pihak berwenang.

2) **Pasien**

Kebijakan-kebijakan dari pasien adalah sebagai berikut:

- Pasien dapat *cancel* antrian pada satu jam sebelum dokter praktek

3) **Asisten Dokter(Perawat)**

Kebijakan-kebijakan dari asisten dokter adalah sebagai berikut:

- Asisten dokter dapat menutup antrian atau menghentikan antrian pada satu 30menit sebelum antrian dibuka.
- Asisten dokter dapat menambah jadwal praktek dadakan pada satu jam sebelum jadwal tersebut.

6.1.1.3 Analisis Kebutuhan Sistem

6.1.1.3.1 Aturan Sistem Antrian

Berikut adalah aturan sistem antrian yang baru yang harus diperhatikan bagi pengguna yang mempunyai akun. Aturan yang berlaku tidak boleh dilanggar oleh semua pengguna dan menjalankan wewenang yang berlaku dengan baik. Berikut adalah aturan-aturan dari sistem antrian yang baru:

- Pengguna yang mempunyai akses masuk ke dalam sistem antrian harus menjaga dan tidak melakukan penyebaran *username* dan *password* kepada orang lain.
- Pengguna dilarang menggunakan akun pengguna lain.
- Pengguna yang mempunyai akses masuk sistem antrian dilarang mempublikasikan atau menyebarkan informasi yang ada di sistem antrian kepada orang lain melalui media apapun.

6.1.1.3.2 Kebutuhan Fungsional

Menggambarkan fungsionalitas sistem atau layanan-layanan sistem. Kebutuhan fungsional didapatkan dari hasil wawancara yang dilakukan kepada pegawai, dan mendapatkan beberapa kebutuhan fungsional yaitu :

Tabel 4.2 Kebutuhan Fungsional

KF-01	Sistem menyediakan fitur bagi pengguna untuk <i>login</i> dan <i>logout</i>
KF-02	Sistem menyediakan fitur bagi admin untuk mengelola data pengguna (pasien) dan data dokter
KF-03	Sistem menyediakan fitur bagi admin untuk mengelola jadwal dokter (jam, hari, batas maksimal kesanggupan dokter)(default)
KF-04	Sistem menyediakan fitur bagi asisten dokter(perawat) untuk melakukan update antrian
KF-05	Sistem menyediakan fitur untuk menampilkan daftar antrian dan sisa antrian
KF-06	Sistem menyediakan fitur bagi pasien untuk pengambilan antrian

6.1.1.3.3 Kebutuhan Non Fungsional

Pada tahapan ini adalah menginisialisasikan kebutuhan non fungsional dari beberapa aspek. Berikut adalah kebutuhan non fungsional yang diidentifikasi.

Usability Requirement

Usability requirement adalah kebutuhan non fungsional terkait dengan kemudahan penggunaan sistem atau perangkat lunak oleh user. Berikut adalah kebutuhan non fungsional dari *usability*.

Tabel 4.3 Usability Requirement

KNF-01	Semua fitur yang tersedia dapat digunakan sebagaimana fungsinya
KNF-02	Tidak ada menu atau tombol yang membingungkan bagi pengguna

Reliability and Security Requirement

Reliability and security requirement yaitu kebutuhan terkait keandalan sistem atau perangkat lunak termasuk juga faktor keamanan (*security*) sistem. Kebutuhan non fungsional dari *reliability and security* adalah sebagai berikut.

Tabel 4.4 Reliability Requirement

KNF-03	Sistem dapat menampilkan seluruh konten dari sistem antrian
KNF-04	Hanya admin yang dapat mendaftarkan atau menambah pengguna
KNF-05	Hanya admin dan asisten dokter yang dapat mengupdate antrian

Portability Requirement

Portability requirement adalah kebutuhan non fungsional yang terkait dengan kemudahan dalam pengaksesan seperti faktor waktu, lokasi pengaksesan dan perangkat yang digunakan. Perangkat tersebut meliputi perangkat keras, perangkat lunak dan perangkat jaringan.

Tabel 4.5 Portability Requirement

KNF-06	Waktu untuk penanganan pada saat server down maksimal 1x24 jam (1 hari)
--------	---

Supportability Requirement

Supportability requirement adalah kebutuhan non fungsional terkait dengan pendukung dari sistem antrian yang baru. Adapun kebutuhan fungsional terkait *supportability* adalah sebagai berikut.

Tabel 4.6 Supportability Requirement

KNF-07	Sistem dapat diakses di segala OS (operating system) untuk website
KNF-08	Sistem dapat diakses di segala OS Android untuk mobile

6.1.1.3.4 Use case

Use-case yang digunakan pada perancangan sistem antrian yang baru ini adalah hasil dari penggalan kebutuhan yang sebelumnya dilakukan. Berikut adalah usecase yang ada pada sistem antrian yang baru.

Tabel 4.7 Use-Case

	Kode use-case	Use-case	Aktor
KF-01	UC-01.01	Masuk akun	Semua Pengguna
	UC-01.02	Keluar akun	Semua Pengguna
KF-02	UC-02.01	Lihat Daftar Data Pengguna	Administrator
	UC-02.02	Input Data Dokter	Administrator
	UC-02.03	Input Data Pasien	Administrator
	UC-02.04	Edit Data Dokter	Administrator
	UC-02.05	Edit Data Pasien	Administrator
	UC-02.06	Hapus Data Dokter	Administrator
	UC-02.07	Cari Data	Administrator

KF-03	UC-03.01	Menambah jadwal dokter	Administrator
	UC-03.02	Mengubah jadwal dokter	Administrator dan asisten dokter (perawat)
	UC-03.03	Me-non-aktifkan jadwal dokter	Administrator
KF-04	UC-04.01	Monitoring Antrian	Asisten Dokter (perawat)
	UC-04.02	Set Antrian	Asisten Dokter (perawat)
KF-05	UC-05.01	Lihat Antrian	Semua Pengguna
KF-06	UC-06.01	Ambil Antrian	Pasien
	UC-06.02	Lihat waktu sisa antrian	Pasien

6.1.1.3.4.1 Use-case Deskripsi

Tahap ini adalah tahap dimana setiap masing-masing *use-case* dijabarkan dengan cara deskripsi. Didalam deskripsi *use-case* akan dijelaskan alur dari masing-masing *use-case*. Berikut adalah contoh salah satu deskripsi *use-case*.

Tabel 4.8 Use-Case (UCD) Ubah Data Dokter

+	UC-02.04 Edit Data Dokter
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengubah data dokter
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi	Jarang (<i>rarely</i>)
Penggunaan :	
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Data Dokter” 2. Sistem menampilkan “Data Dokter” 3. Aktor menekan icon edit () pada salah satu kolom dokter yang ingin di edit 4. Sistem akan menampilkan halaman “Data Dokter ‘A’” 5. Aktor mengisi ulang data dokter yang ingin diganti “NIK”, “Nama”, “Tempat Tanggal Lahir”, “Alamat”, “Nomer Hp”, “Spesialis”, dan “Alamat Email” <ol style="list-style-type: none"> a. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 b. Apabila <i>field</i> “NIK” sudah terpakai, maka sistem akan menjalankan alternatif I no. 2 6. Aktor menekan tombol

Skenario Alternatif:	<ol style="list-style-type: none"> a. “Save”, maka sistem akan menampilkan halaman pemberitahuan “Penambahan Data Berhasil” b. “Cancel”, maka sistem akan menampilkan halaman “List Data Dokter” <p>7. Sistem akan memperbarui Data Dokter dan kembali ke halaman “List Data Dokter”</p>
	<p>❖ Alternatif I</p> <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 2. Sistem akan menampilkan peringatan (<i>notification</i>), yang berisi bahwa “NIK sudah terpakai”

Untuk lebih lengkapnya dapat dilihat pada Lampiran.

6.1.1.3.4.2 Mengelompokkan Use-case Berdasarkan Area Fungsi

Pengelompokkan *use-case* berdasarkan fungsionalitas bertujuan untuk mengetahui area fungsional dari perangkat lunak sistem antrian yang baru. Pengelompokkan *use-case* berdasarkan fungsionalitas terdiri dari:

- Pengelolaan akun,
- Pengelolaan antrian, dan
- Pengelolaan jadwal.

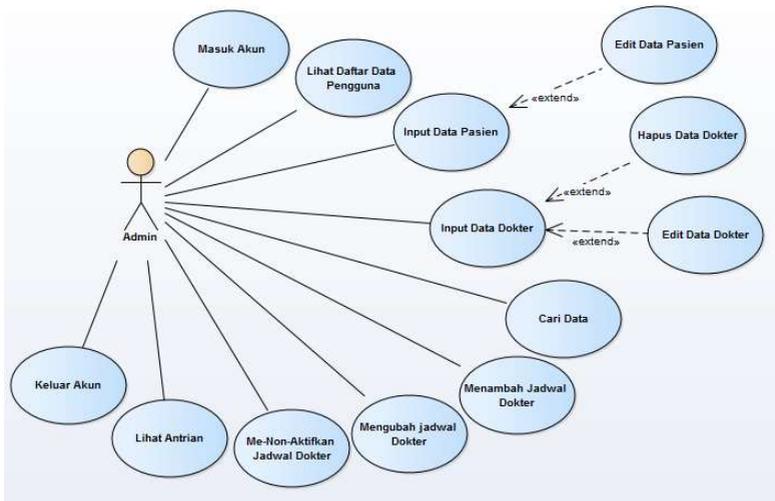
Keterangan dari masing-masing bagian dapat dilihat di dalam lampiran.

6.1.1.3.4.3 Mengelompokkan Use-case Berdasarkan Pengguna

Pengguna-pengguna yang terlibat di dalam perangkat lunak sistem antrian yang baru diantaranya adalah:

- Administrator,
- Pasien, dan
- Asisten Dokter(Perawat).

Pada Gambar berikut ini adalah diagram *use-case* untuk administrator



Gambar 18 Use-Case Diagram Administrator

Untuk keterangan dari masing-masing pengguna dan juga diagram *use-case* selengkapnya ada pada lampiran.

6.1.2 Desain

Pada proses desain ini bentuk dari diagram sistem antrian dan interface dari perangkat lunak sistem antrian yang baru akan terlihat. Proses desain ini menjadi penting karena tugas akhir ini tidak melakukan rancang bangun perangkat lunak, tetapi hanya mendesain sistem antrian yang baru dengan menggunakan *tools GUI Design Studio*. Untuk pembuatan diagram menggunakan *Microsoft Visio*, *Power Designer*, dan *Enterprise Architect*.

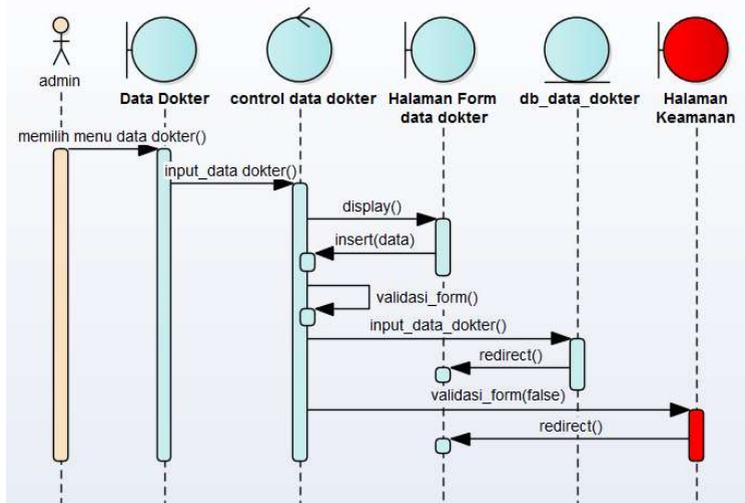
Masukkan dari proses desain ini adalah dokumen spesifikasi perangkat lunak, sedangkan hasil dari proses desain ini adalah validasi desain terhadap kebutuhan fungsional sistem. Pada proses desain ini dibagi menjadi 2, yaitu desain perilaku dan desain antarmuka.

6.1.2.1 Desain Perilaku

Desain tingkah laku ini adalah pola komunikasi antara objek dan relasinya. Tujuannya adalah untuk melihat hubungan interaksi antara pengguna dan objek bisnis dengan sistem.

6.1.2.1.1 Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan jalannya sistem saat digunakan oleh aktor. Untuk lebih lengkapnya penggambaran dari *Sequence Diagram* dapat dilihat di lampiran.



Gambar 19 Sequence Input Data Dokter

Untuk lebih lengkapnya dapat dilihat pada Lampiran.

6.1.2.1.2 Class Diagram

Class diagram adalah diagram yang menggambarkan *method* atau kelas untuk program yang akan dibuat. Untuk hasil dari *class diagram* yang lengkap dapat dilihat dilampiran.



Gambar 20 Class Diagram Admin

6.1.2.2 Desain Antarmuka

Desain antarmuka digunakan untuk menjelaskan dari gambaran dari tampilan sistem antrian yang dibangun. Untuk lebih detailnya akan dijelaskan pada lampiran.

6.1.2.2.1 Website

Dibawah ini adalah beberapa desain antarmuka atau User Interface website yang diperuntukkan untuk administrator dan asisten dokter

The image shows a login interface for a website. At the top, there is a banner with the word 'ANTRI' in large, colorful letters (A: red, N: yellow, T: green, R: blue, I: purple). Below the banner is a red 'Log in' button. Underneath are two input fields: 'USERNAME' and 'PASSWORD'. Below these fields are two radio buttons: 'Administrasi' (selected) and 'Asisten Dokter'. At the bottom, there is a 'Log In' button.

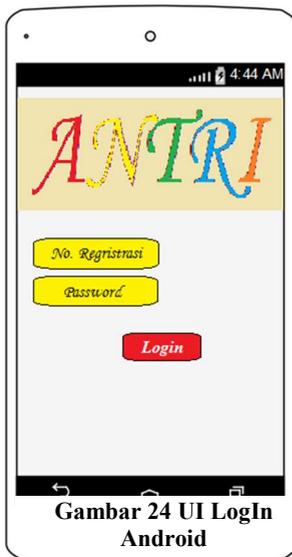
Gambar 21 user interface Login website

yang ada pada aplikasi ANTRI. Untuk lebih lengkap lihat pada lampiran.

Gambar 22 User Interface Tambah Data Pasien

6.1.2.2.2 Mobile Android

Dibawah ini adalah beberapa desain antarmuka atau User Interface website yang diperuntukkan untuk administrator dan asisten dokter yang ada pada aplikasi ANTRI. Untuk lebih lengkap lihat pada lampiran.



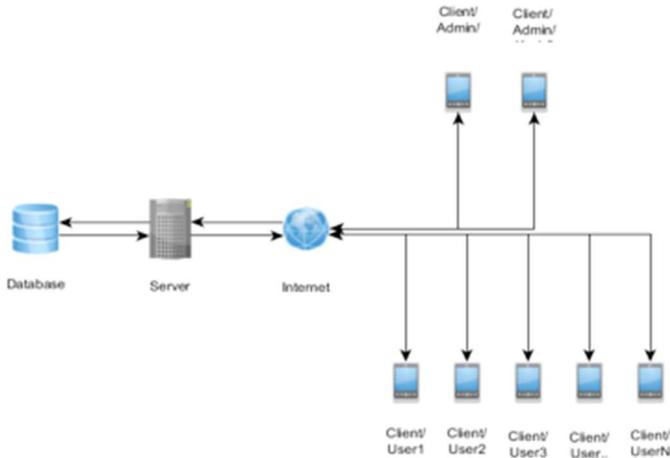
**Gambar 24 UI LogIn
Android**



**Gambar 23 UI Jadwal
Dokter Android**

6.1.2.3 Desain Arsitektur

Aplikasi “ANTRI” terdiri dari client dan server, client digunakan untuk mengakses, merubah dan memperbarui informasi seputar antrian melalui smartphone dan komputer. Server digunakan sebagai tempat penyimpanan data antrian online yang bisa diakses oleh client dengan menggunakan smartphone.



Gambar 25 Design Arsitektur Aplikasi

Gambar diatas merupakan arsitektur komunikasi Aplikasi “ANTRI” untuk mendaftar atau melihat data antrian dari aplikasi kedalam database yang ada didalam server. Untuk mendaftarkan diri atau mendapatkan informasi mengenai antrian yang terjadi di bagian administrasi, pasien diharuskan menginstal aplikasi “ANTRI” pada smartphone. Dan untuk pihak rumah sakit mengaksesnya melalui URL.

6.1.3 Validasi

Validasi bertujuan untuk memastikan bahwa system aplikasi yang dibangun sudah sesuai, yaitu produk yang memenuhi tujuan spesifik yang diharapkan. Peneliti menggunakan matrik kerunutan sebagai cara untuk memvalidasi desain aplikasi.

6.1.3.1 Matriks Keruntutan

Menjelaskan tentang keruntutan dari kesesuaian antara kebutuhan fungsional (KF), *Use-case (UC)*, *Use-case Scenario (UCS)*, *Sequence Diagram (SD)*, dan *Interface*. Dengan menggunakan matriks keruntutan, maka dapat diketahui apakah desain yang dibuat, telah sesuai dengan kebutuhan fungsional maupun *use-case* yang dibuat diawal.

Tabel 6.1 Mariks Kerunutan

Kebutuhan Fungsional	Kode use-case	Use-case	Sequence Diagram (Lampiran DDPL)	Interface Admin (SKPL)	Interface AssDok (SKPL)	Interface Pasien (SKPL)
KF-01	UC-01.01	Masuk akun	Gambar 4 SD LogIn	Gambar 9 UI Login Admin dan AssDok	Gambar 9 UI Login Admin dan AssDok	Gambar 1 Login Pasien
	UC-01.02	Keluar akun	Gambar 5 SD LogOut	Gambar 10 UI Halaman Awal Admin (Setiap Halaman)	Gambar 23 UI Monitoring Antrian (Setiap Halaman)	Gambar 2 (Setiap Halaman)
KF-02	UC-02.01	Lihat Daftar Data Pengguna	Gambar 6 SD Lihat Daftar Pengguna	Gambar 12 Daftar Data Pengguna	-	-

	UC-02.02	Input Data Dokter	Gambar 11 SD Input Data Dokter	Gambar 15 UI Data Dokter	-	-
	UC-02.03	Input Data Pasien	Gambar 12 SD Input Data Pasien	Gambar 13 UI Input Data Pasien	-	-
	UC-02.04	Edit Data Dokter	Gambar 8 SD Edit Data Dokter	Gambar 16 UI Edit Data Dokter	-	-
	UC-02.05	Edit Data Pasien	Gambar 9 SD Edit Data Pasien	Gambar 14 UI Edit Data Pasien	-	-
	UC-02.06	Hapus Data Dokter	Gambar 10 SD Hapus Data Dokter	Gambar 17 UI Hapus Data Dokter	-	-
	UC-02.07	Cari Data	Gambar 7 SD Cari Data	Gambar 18 UI Cari Data	-	-
KF-03	UC-03.01	Menambah jadwal dokter	Gambar 13 SD Tambah	Gambar 20 UI Tambah	-	-

			Jadwal Dokter	Jadwal Dokter		
	UC-03.02	Mengubah jadwal dokter	Gambar 14 SD Mengubah jadwal dokter	Gambar 21 UI Ubah Jadwal Dokter	-	-
	UC-03.03	Me-non-aktifkan jadwal dokter	Gambar 15 SD Menonaktifkan Jadwal Dokter	Gambar 22 UI Menonaktifkan Jadwal Dokter	-	-
KF-04	UC-04.01	Monitoring Antrian	Gambar 16 SD Monitoring	-	Gambar 23 UI Monitoring Antrian	-
	UC-04.02	Set Antrian	Gambar 17 SD Set Antrian	-	Gambar 24 UI Set Antrian	-
	UC-04.03	Panggil Antrian	Gambar 18 SD Antrian	-	Gambar 26 UI Panggil Antrian	-

	UC-04.04	Next Antrian	Gambar 18 SD Antrian	-	Gambar 25 UI Next Antrian	-
KF-05	UC-05.01	Lihat Antrian	Gambar 19 SD Lihat Antrian	Gambar 27 UI Lihat Antrian Admin	Gambar 28 UI Lihat Antrian AssDok	Gambar 1 UI Home Pasien
KF-06	UC-06.01	Ambil Antrian	Gambar 20 SD Ambil Antrian	-	-	Gambar 2 UI Home Pasien
	UC-06.02	Antrian Saya	Gambar 21 SD Antrian Saya	-	-	Gambar 5 UI Antrian Saya

(Halaman ini sengaja dikosongkan)

BAB VII

KESIMPULAN DAN SARAN

Bab ini akan menjelaskan kesimpulan dari penelitian ini, beserta saran yang dapat bermanfaat untuk perbaikan di penelitian selanjutnya.

7.1 Kesimpulan

Simpulan yang dapat diambil dari pengerjaan tugas akhir ini adalah sebagai berikut :

1. Penggalan kebutuhan dilakukan dengan melakukan wawancara kepada pihak RSIA Pondok TJandra Wawancara dilakukan dengan dua kali wawancara.
2. Wawancara pertama fokus pada proses bisnis secara umum dan identifikasi kondisi eksisting dari RSIA. Wawancara kedua fokus kepada pengguna dari aplikasi nantinya.
3. Kebutuhan fungsional dan *use-case* dari Aplikasi ANTRI, didapatkan dari hasil wawancara dan *benchmarking*.
4. Ada *use-case* yang dapat digunakan oleh semua aktor (pengguna) seperti : lihat antrian.
5. Terdapat beberapa *use-case* yang tidak memiliki kebutuhan fungsional.
6. Terdapat beberapa *use-case* yang memiliki lebih dari satu kebutuhan fungsional.
7. Desain *Database* memiliki 5 tabel (*class*) yaitu : Pasien, Dokter, Asisten Dokter, Jadwal dokter Dan Antrian.
8. Hasil dari perancangan desain menggunakan *GUI Design Studio*.

Validasi desain dilakukan dengan membuat matrik keruntutan.

Hasil validasi ini adalah semua kebutuhan yang diperoleh dari wawancara dan *benchmarking* telah sesuai

7.2 Saran

Pada bagian ini peneliti akan memberikan saran dari hasil analisis dan perancangan aplikasi “ANTRI”. Saran yang diharapkan dapat dikembangkan di masa mendatang adalah:

1. Peneliti yang ingin meneliti lebih lanjut tentang sistem antrian dapat melihat dari sistem antrian dari pelayanan lainnya (apotek atau laboratorium) agar sistem antrian dan aplikasi dari sistem antrian bisa berjalan dengan optimal dan efisien.
2. Pihak RSIA atau pengembang dapat membuat aplikasi sistem antrian tersebut dengan dokumen yang sudah dibuat oleh peneliti untuk menggantikan sistem antrian yang masih manual dengan sistem antrian yang baru untuk meningkatkan pelayanan RS yang lebih baik lagi

DAFTAR PUSTAKA

- [1] 27 januari 2017. [Online]. Available: https://id.wikipedia.org/wiki/Rumah_sakit.
- [2] 09 Maret 2017. [Online]. Available: <http://www.depkes.go.id/resources/download/peraturan/UU%20No.%2044%20Th%202009%20ttg%20Rumah%20Sakit.PDF>.
- [3] P. Zave, Classification of Research Efforts in Requirements Engineering, ACM Computing Surveys, 29(4), pp. 315-32, 1997.
- [4] r. s. wahono, 08 Maret 2017. [Online]. Available: <http://romisatriawahono.net/2006/04/29/menyegarkan-kembali-pemahaman-tentang-requirement-engineering/>.
- [5] Z. Hakim, 08 Maret 2017. [Online]. Available: <http://www.zainalhakim.web.id/pengertian-dan-langkah-langkah-desain-sistem.html#sthash.VNxRYAtG.dpuf>.
- [6] H. Al Fatta, Analisis dan Perancangan Sistem informasi untuk keunggulan bersaing perusahaan & Organisasi Modern, Yogyakarta: Andi, 2007.
- [7] M. Fowler and K. Scoot, UML Distilled Second Edition A Brief Guide to the Standard Object Modeling Language, Addison Wsley, 1999.
- [8] P. Roger, Rekayasa Perangkat Lunak, Yogyakarta: Andi, 2012.
- [9] P. P.-S. Chen, "English Setence Structure and Entity-Relationship Diagrams," Elsevier Science Publishing, New York, 1983.

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS



Nurul Fakhria lahir di Sidoarjo, 12 September 1991. Penulis adalah putri terakhir dari empat bersaudara dan sejak kecil tinggal di kota Sidoarjo, Jawa Timur. Pendidikan formal yang pernah ditempuh oleh penulis dimulai dari MI NU Ngingas Waru-Sidoarjo (1997), MTs Darul Ulum Kureksari, Waru-Sidoarjo (2003-2006), SMAN 1 Waru-Sidoarjo (2006-2009), setelah itu penulis melanjutkan Pendidikan ke tahap perguruan tinggi di Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Surabaya angkatan 2009 dengan NRP 52 09 100 097.

Di program Studi Sistem Informasi ini Penulis mengambil Bidang Minat Manajemen Sistem Informasi (MSI). Penulis dapat dihubungi melalui *e-mail* nurul.fakhria@gmail.com.

(halaman sengaja dikosongkan)

LAMPIRAN A – HASIL WAWANCARA

Tugas Akhir :
“ANALISIS DAN PERANCANGAN APLIKASI “ANTRI”
BERBASIS WEBSITE DAN MOBILE ANDORID PADA RSIA
PONDOK TJANDRA

Tanggal : 17 April 2017
Lokasi : RSIA PONDOK TJANDRA
Pewawancara : Nurul Fakhria
Narasumber/jabatan : Imatus Solichah (Staf Administrasi Umum)

1. Apakah posisi Anda di RSB Pondok Tjandra?
∴ Saya sebagai staf administrasi
2. Sudah berapa lama Anda bekerja di bagian ini?
∴ Saya bekerja sebagai administrasi disini sejak tahun 2015, 2 tahun yang lalu
3. Bisa Anda ceritakan mengenai alur bisnis RSB saat ini?
Aktivitas apa saja yang ada pada RSB?
∴ Di rumah sakit RSB Pondok Tjandra ini melayani pemeriksaan kehamilan, melahirkan, imunisasi, pemeriksaan anak dan ada dokter umum yang berjaga di UGD. Jadi disini hanya ada 2 poli, poli kandungan dan anak. Dan ada beberapa dokter di tiap-tiap poli, dengan jadwal praktek yang berbeda-beda. Dengan keterbatasan ruangan, jadi tidak ada dokter di poli yang sama dengan jadwal praktek yang sama. Itu juga termasuk solusi penanganan disini, diusahakan tiap hari ada dokter praktek dimasing-masing poli. Karena ini rumah sakit swasta yang didirikan oleh dr. Supri, jadi bisa dipastikan kebanyakan pasien memilih dokter supri. Sedangkan beberapa dokter

juga mempunyai cabang praktek yang lain. Jadi tidak semua tercover ke dokter supri. Dan ada yang memilih dokter lain. Disini pasien baru yang belum pernah periksa disini, daftar dulu mengisi formulir lalu diserahkan kepada saya atau teman yang lain. Lalu saya masukan ke microsoft excel agar mudah mencari rekam medik dilaci mana. Pasien akan mendapatkan nomer registrasi yang akan di sebutkan atau dipakai untuk daftar periksa selanjutnya. Setelah itu pasien dapat nomer antriannya dan menunggu tiba masuk. Setelah selesai pemeriksaan, menuju administrasi untuk membayar biaya periksa. Jika ada resep yang harus ditebus, maka pasien menuju apotek untuk menebus obat, dan langsung membayar biaya obat tersebut, dan menunggu obat. Jika harus menjalani tes darah, maka pasien membawa surat pengantar ke bagian laboratorium dan membayar biaya kemudian melakukan tes.

4. Sudah pernah di lakukan pengembangan IT? Sistem apa yang didukung IT?

∴. Disini hanya menggunakan microsoft excel untuk memasukkan data pasien yang digunakan untuk mencari rekam medik dengan nomer laci, kemudian kita mengambil kertas rekam medik tersebut, yang disetorkan pada dokter saat periksa. Di bagian apotek juga dilakukan pencatatan/rekapan keluar masuknya obat dengan m. Excel ketika diakhir shift masing-masing.

5. Bagaimana prosedur pendaftaran pemeriksaan pasien(antrian)?

∴. Pasien bisa mendaftar langsung atau melalui telepon, kebanyakan melalui telepon. Dengan menyebutkan nomer registrasi dan kami mencatat dibuku antrian dan di berikan nomer urut periksa. Seringnya pasien menanyakan jam berapa dokter datang atau mulai praktek. Karena kadang dokter sedang menangani pasien caesar, jadi beberapa meleset dari jam mulai praktek.

6. Apakah prosedur pendaftaran pemeriksaan pasien saat ini terdapat kendala?

∴ Ya kendalanya sih karena jadwal praktek yang berubah-ubah yang menjadikan pasien mungkin tidak nyaman. Tetapi sejauh ini tidak ada kendala tentang nomer antrian yang sama, karena buku antrian standby di meja administrasi. Jadi kami bisa melihatnya berkali-kali.

7. Jika ada, bagaimana cara menanganinya?

∴ Menanganinya ya dengan memberi penjelasan saja kepada pasien. Beberapa pasien ada yang meninggalkan sebentar, dan seringnya pasien mengandalkan telepon untuk mendapat informasi dokter sudah datang atau belum, sudah mulai praktek atau belum, sudah sampai nomer berapa. Tapi jika praktek dokter maju dari jadwal atau sepi pasien nomer awal, biasanya kita hubungin pasien dengan pendaftaran terdekat untuk menginformasikan agar segera hadir. Jika nomer awal belum hadir ya terpaksa kita lompati ke nomer terdekat yang sudah hadir.

8. Jika dilakukan Antrian berbasis IT, apa harapannya?

∴ Harapannya ya agar antrian disini teratur, dan tidak ada kesalahpahaman antrian. Mungkin bisa membuat pasien puas dengan pelayanan kami.

LAMPIRAN B – WAWANCARA DENGAN PASIEN

Tugas Akhir :
“ANALISIS DAN PERANCANGAN APLIKASI “ANTRI”
BERBASIS WEBSITE DAN MOBILE ANDORID PADA RSIA
PONDOK TJANDRA

➤ IDENTITAS PASIEN

Nama : Anita
Umur : 32 tahun

1. Sudah berapa lama bapak/ibu/saudara/saudari menjadi pasien di Rumah Sakit Pondok Tjandra?
∴. Sudah 7tahun, sejak hamil anak pertama
2. Apa yang bapak/ibu/saudara/saudari rasakan terhadap pelayanan rumah sakit ini?
∴. Sejauh ini pelayanan rumah sakit secara keseluruhan sudah baik.
3. Bagaimana alur pendaftaran antrian pemeriksaan dokter?
∴. Biasanya saya telepon dulu untuk menanyakan jadwal praktek dokternya, kemudian jika saya berminat dan bisa saat itu ya saya langsung daftar. Biasanya pihak administrasi menanyakan nomer regristrasi. Kita tinggal menyebutkannya saja, dan petugas akan memberitahukan nomer antrian kita.
4. Apakah sudah optimal? Apa ada kendala yang terjadi pada saat pendaftaran?
∴. Kalau menurut saya sudah, karena sampai saat ini saya lancar-lancar saja. Tapi kadang ada juga yang beberapa pasien yang terselip namanya, jadi tidak di panggil2.

5. Jika ada, apa yang bapak/ibu/saudara/saudari harapkan dengan permasalahan tersebut?

∴. Harapannya sih masalah-masalah terhadap antrian sudah tidak ada. Karena disini rumah sakit ibu dan anak jadi hampir semua pasien-pasiennya adalah anak-anak dan para wanita, terutama ibu hamil yang sangat membutuhkan kenyamanan.

6. Bagaimana jika ada Aplikasi untuk mengambil nomer antrian? Apa Anda setuju? Jadi bisa mengambil nomer antrian dari mana saja.

∴. Setuju sekali, biar tidak penuh ruang tunggu.

LAMPIRAN C – WAWANCARA KE-DUA

Pertanyaan Baru Tentang Pengguna (Dengan Staf Administrasi)

1. Siapa saja user pengguna dari aplikasi yang akan dibuat ?
∴ Fungsi dari aplikasi ini nantinya akan menggantikan sistem antrian kita yang lama atau manual, jadi pengguna nya adalah:
 - *Staf Administrasi, yang mengelola data-data pasien dan dokter dan bertindak sebagai admin pusat dari aplikasi tersebut.*
 - *Assisten Dokter atau perawat yang membantu dokter pada saat pemeriksaan.*
 - *Dan Pasien, sebagai klien kita.*
2. Bagaimana pembagian *job desk* dari pengguna aplikasi ?
∴ Nantinya perlu adanya penyaringan informasi yang dilakukan oleh staf dari bidang administrasi agar informasi yang ada di aplikasi nantinya tidak sering mengalami perubahan-perubahan. Terutama pada pihak admin dan asisten doter
3. Apakah nantinya diperlukan adanya pendaftaran akun masing-masing ?
∴ Untuk mengantisipasi hal-hal yang tidak memungkinkan. Lebih baik kalau akun Login sudah kami atur, jadi nanti para pengguna bisa lang sung login tanpa perlu daftar.

LAMPIRAN D – WAWANCARA KE-TIGA

Pertanyaan Baru Tentang Desain (New Questions About Design)

1. Apakah tampilan aplikasi “ANTRI” ini telah sesuai dengan kebutuhan / keinginan anda ?
∴. Untuk tampilan aplikasi ini, saya rasa telah memenuhi apa yang telah saya sampaikan tempo hari (saat interview penggalan kebutuhan).
2. Bagaimana pendapat anda mengenai tampilan awal dari aplikasi ini ?
∴. Untuk tampilan ini, saya rasa sudah cukup bagus tampilannya.
3. Bagaimana pendapat anda mengenai tampilan desain dan warna pada aplikasi ini ?
∴. Untuk tampilan dan kolaborasi warna saya rasa sudah cukup karena tidak terlalu ramai, tetapi mungkin nantinya ada hiasan berupa gambar atau Logo RS
4. Bagaimana pendapat anda mengenai tampilan dari setiap usernya ?
∴. Tampilan untuk setiap user/actor, tidak membingungkan dengan kata lain meski aktornya nantinya adalah orang yang tidak mengerti tentang teknologi sama sekali, sepertinya tidak akan kesulitan untuk mengoperasikan setiap menu dari aplikasi ini, karena bahasa yang digunakan menggunakan bahasa Indonesia.
5. Menurut anda apakah desain aplikasi ini nantinya tidak membingungkan user nantinya ?

∴ Saya rasa tidak akan membingungkan, dikarenakan tidak ada menu yang membingungkan (tidak ada ambiguitas) dari tampilan aplikasi ini, dan secara overall tampilan ini sudah sangat bagus dan actor yang menggunakan nantinya tidak akan merasa kebingungan dalam pengoperasiannya.

6. Apakah menurut anda ada yang masih kurang dari aplikasi ini ?

∴ Secara keseluruhan saya, sudah cukup puas dan saya rasa tidak ada yang nantinya akan membingungkan bagi pengguna orang awam sekalipun. Tetapi yang perlu di perhatikan nantinya tidak ada menu atau fitur atau tombol yang bersifat membingungkan (ambiguitas). Apabila ada kerusakan (maintenance) server bisa segera di tangani tidak lebih dari 1 hari, karena dapat mengganggu fungsi dan tujuan dari aplikasi ini.

LAMPIRAN E – SKPL



SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK



Disiapkan:

Nurul Fakhria 5209100097

sebagai kelengkapan Tugas Akhir di Jurusan Sistem Informasi
ITS

Abstrak:

Dokumen ini berisi tentang Spesifikasi Perangkat Lunak pada
Aplikasi ANTRI



SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK



Disiapkan:

Nurul Fakhria 5209100097

sebagai kelengkapan Tugas Akhir di Jurusan Sistem Informasi ITS

Abstrak:

Dokumen ini berisi tentang Spesifikasi Perangkat Lunak pada Aplikasi ANTRI

Versi Dokumen

Seluruh versi dari dokumen ini didaftar berdasar kronologisnya. Tidak ada hubungan antara nomer dokumen dan nomor versi perangkat lunak.

Tabel 9 Versi Dokumen

Versi Dokumen	Tanggal	Alasan Perubahan	Versi Perangkat Lunak
DRAF	15-06-2017		
Release 1	15-06-2017		

Dokumen ini dibuat oleh Nurul Fakhria dengan pengawasan dari Jurusan Sistem Informasi ITS sebagai upaya untuk menjamin keakurasian dokumen saat akan di cetak. Penggandaan dokumen, sebaiknya dari versi yang terakhir (up to date) dan setelah mendapatkan ijin tertulis

Copyright @ 2006 Jurusan Sistem Informasi ITS
Seluruh informasinya adalah hak milik Jurusan Sistem Informasi ITS yang tidak dipublikasikan dan bersifat rahasia.

Revisi Dokumen

Seluruh revisi yang telah dilakukan pada dokumen ini, dapat diikuti sebagaimana tabel berikut.

Tabel 10 Revisi Dokumen

Nomer Revisi	Tanggal	Diperiksa oleh	Keterangan singkat perbaikan

Daftar Isi

Daftar Isi.....	iii
Daftar Tabel.....	v
Daftar Gambar.....	vii
1. Pendahuluan.....	1
1.1. Tujuan.....	1
1.2. Ruang Lingkup Perangkat Lunak.....	1
1.3. Target Audience	1
1.4. Definisi, Istilah dan Singkatan	1
1.5. Sistem Penomoran.....	2
1.6. Referensi.....	2
1.7. Ikhtisar Dokumen.....	3
2. Deskripsi Umum.....	3
2.1. Tentang Perangkat Lunak.....	3
2.2. Fungsi-fungsi Perangkat Lunak.....	3
2.3. Karakteristik dan Klasifikasi Pengguna	4
2.4. Batasan Desain dan Implementasi.....	5
2.5. Dokumentasi Bagi Pengguna	5
2.6. Asumsi dan Ketergantungan	5
3. Kebutuhan Antarmuka Eksternal.....	6
3.1. Antarmuka Pengguna	6
3.2. Antarmuka Perangkat Keras.....	19
3.3. Antarmuka Perangkat Lunak.....	19
3.4. Antarmuka Komunikasi	19
4. Feature Sistem	20
5. Kebutuhan NonFungsional Lainnya	46
5.1. Usability requirement.....	46
5.2. Reliability and security requirement	47

5.3. Portability Requirement	47
5.4. Supportability requirement.....	47
6. Kebutuhan Lain	47
Appendix A: Model Analisis.....	48
Appendix B: Daftar Kebutuhan.....	49

Daftar Tabel

Tabel 1 Versi Dokumen	i
Tabel 2 Revisi Dokumen	ii
Tabel 3 Sistem Penomoran.....	2
Tabel 4 Kebutuhan Fungsional.....	20
Tabel 5 Use-case	23
Tabel 6 Use-Case Pengelolaan Akun	24
Tabel 7 Use-Case Pengelolaan Antrian.....	24
Tabel 8 Use-Case Pengelolaan Jadwal.....	24
Tabel 9 Use-Case Use-Case Administrator.....	25
Tabel 10 Use-Case Asisten Dokter	25
Tabel 11 Use-Case Pasien	25
Tabel 12 UCD Masuk Akun(Login)	26
Tabel 13 UCD Keluar Akun.....	26
Tabel 14 UCD Lihat Daftar Data Pengguna.....	27
Tabel 15 UCD Input Data Dokter	29
Tabel 16 UCD Input Data Pasien	30
Tabel 17 UCD Edit Data Dokter	31
Tabel 18 UCD Edit Data Pasien.....	33
Tabel 19 UCD Hapus Data Dokter.....	35
Tabel 20 UCD Cari Data	36
Tabel 21 UCD Tambah Jadwal Dokter	37
Tabel 22 UCD Ubah Jadwal Dokter.....	38
Tabel 23 UCD Menonaktifkan Jadwal Dokter.....	39
Tabel 24 UCD Monitoring Antrian	40
Tabel 25 UCD Set Antrian	40
Tabel 26 UCD Panggil Antrian	41
Tabel 27 UCD Next Antrian	43
Tabel 28 UCD Lihat Antrian.....	43
Tabel 29 UCD Ambil Antrian	44
Tabel 30 UCD Antrian Saya	46
Tabel 31 KNF Usability	46

Tabel 32 KNF Reliability and Security	47
Tabel 33 KNF Portability	47
Tabel 34 KNF Supportability	47

Daftar Gambar

Gambar 1 UI Login Pasien.....	6
Gambar 2 UI Home Pasien.....	6
Gambar 3 UI Ambil Antrian	6
Gambar 4 UI Antrian.....	6
Gambar 5 UI Antrian Saya.....	7
Gambar 6 UI Waktu Antrian	7
Gambar 7 Ui Tidak Ada Antrian.....	7
Gambar 8 UI Ada Antrian.....	7
Gambar 9 UI Login Admin dan AsDok	8
Gambar 10 UI Halaman Awal Admin.....	8
Gambar 11 UI Data Pasien.....	9
Gambar 12 Daftar Data Pengguna.....	9
Gambar 13 UI Input Data Pasien	10
Gambar 14 UI Edit Data Pasien	10
Gambar 15 UI Data Dokter	11
Gambar 16 UI Edit Data Dokter.....	11
Gambar 17 UI Hapus Data Dokter	12
Gambar 18 UI Cari Data	12
Gambar 19 UI Jadwal Dokter.....	13
Gambar 20 UI Tambah Jadwal Dokter.....	13
Gambar 21 UI Ubah Jadwal Dokter	14
Gambar 22 UI Menonaktifkan Jadwal Dokter	14
Gambar 23 UI Monitoring Antrian	16
Gambar 24 UI Set Antrian.....	16
Gambar 25 UI Next Antrian.....	17
Gambar 26 UI Panggil Antrian	17
Gambar 27 UI Lihat Antrian Admin	18
Gambar 28 UI Lihat Antrian AssDok	18
Gambar 29 UCD Admin.....	21
Gambar 30 UCD AssDok.....	21
Gambar 31 UCD Pasien	22

Gambar 32 Arsitektur Sistem	48
Gambar 33 Arsitektur Sistem lain	48

1. Pendahuluan

1.1. Tujuan

Tujuan dari dokumen ini adalah mencatat berbagai kebutuhan fungsional. Kebutuhan fungsional diambil dari kebutuhan pengguna terhadap fungsi-fungsi yang ada didalam sistem, aturan-aturan bisnis beserta atribut kualitas yang mengiringinya. Pada masing-masing kebutuhan fungsional, dilakukan analisis terhadap input, proses (normal dan alternatif), dan output.

1.2. Ruang Lingkup Perangkat Lunak

Ruang lingkup pada perangkat lunak ini adalah :

- Terdapat 3 user, yaitu pasien, administrator (Bagian Administrasi), dan Asisten Dokter.
- Semua user memiliki profil mereka sendiri dalam database
- Pasien dapat mencari jadwal dokter dan mengantri.
- Asisten Dokter dapat membuka dan mengatur antrian.

1.3. Target Audience

Dokumen ini dapat digunakan oleh:

- Pihak Rumah Sakit untuk memahami aplikasi
- Programmer untuk dijadikan acuan pembuatan aplikasi
- dan Mahasiswa yang ingin menjadikan bahan referensi pengerjaan SKPL proyek lain atau untuk melanjutkan pembuatan aplikasi

1.4. Definisi, Istilah dan Singkatan

Guna memberikan gambaran yang sama terhadap beberapa definisi, istilah dan singkatan yang digunakan di dokumen ini, perlu dijelaskan sebagaimana berikut:

IEEE	: Institute of Electrical and Electronics Engineers Adalah standart internasional untuk pengembang perangkat lunak
SKPL	: Spesifikasi Kebutuhan Perangkat Lunak atau
SRS	: Software Requirement Specification

Adalah dokumen hasil analisis sebuah perangkat lunak yang berisi spesifikasi kebutuhan pengguna

1.5. Sistem Penomoran

Ada beberapa hal/bagian dalam dokumen ini yang perlu diberi nomor. Maksud penomoran ini untuk mempermudah audience dalam pengidentifikasian. Adapun aturan penomorannya sebagaimana tabel berikut:

Tabel 11 Sistem Penomoran

Hal/Bagian	Aturan Penomoran
Tabel/Data Store	Nomor berbentuk TD99, dimana 99 adalah nomor urut tabel atau data store Contoh: TD11, TD12, dan sebagainya
Kebutuhan Fungsional	Nomor berbentuk KF999.x, dimana 999 adalah nomor urut struktur butir-butir pada kebutuhan fungsional. Sedangkan x adalah nomor berupa abjad dan sifatnya sebagai tambahan jika kebutuhan fungsional tersebut memiliki item turunannya. Contoh: KF101, KF120.a, dan sebagainya
Kebutuhan Non Fungsional	Nomor berbentuk KnF99.x, dimana 99 adalah nomor urut struktur butir-butir pada kebutuhan non fungsional. Sedangkan x adalah nomor berupa abjad dan sifatnya sebagai tambahan jika kebutuhan non fungsional tersebut memiliki item turunannya. Contoh: KnF11, KnF12.b dan sebagainya

1.6. Referensi

Berikut adalah daftar acuan yang digunakan dalam pendokumentasian spesifikasi kebutuhan perangkat lunak ini.

- IEEE Std. 1233, 1998 Edition IEEE Guide for Developing System Requirements Specifications
- IEEE, Software Requirements Engineering, Second Edition, IEEE Computer Society Press, 2002.
- Bray, Ian K. An Introduction to Requirement Engineering, 1st published, Addison-Wesley, 2002

- Kotonya, Gerald and Sommerville, Ian. Requirement Engineering: Processes and Techniques, John Wiley & Sons Ltd, 1998
- Holil, Achmad. Template: Spesifikasi Kebutuhan Perangkat Lunak, Jurusan Sistem Informasi ITS, 2006.
- Wawancara dengan bagian administrasi rumah sakit.

1.7. Ikhtisar Dokumen

Dokumen ini juga menjelaskan use case dari aplikasi ANTRI secara global, asumsi dalam pengembangan, kebutuhan spesifik dan kebutuhan tambahan.

2. Deskripsi Umum

2.1. Tentang Perangkat Lunak

Aplikasi ANTRI bertujuan untuk menggantikan proses pengambilan nomor antrian untuk pendaftaran pemeriksaan dokter yang semula dilakukan secara manual, menjadi berbasis aplikasi. Dengan adanya aplikasi ini diharapkan proses pengambilan nomor antrian menjadi lebih mudah dan dapat dilakukan dari mana saja selama terkoneksi dengan internet. Dan untuk meningkatkan kualitas pelayanan terhadap pasien. Aplikasi ini terdapat dua tampilan yaitu berbasis website yang di peruntukkan Administrator dan Asisten Dokter dan berbasis Mobile untuk pasien.

2.2. Fungsi-fungsi Perangkat Lunak

Aplikasi ANTRI berfungsi untuk menggantikan proses antri konvensional pada pendaftaran pemeriksaan dokter. Aplikasi ini mempunyai beberapa fungsi:

User (pasien dan asisten dokter) harus sudah terdaftar untuk menggunakan aplikasi ANTRI ini. Apabila belum terdaftar, user dapat mendaftar terlebih dahulu melalui administrator ANTRI, yaitu bagian administrasi umum.

a. Login

User yang sudah terdaftar harus melakukan login terlebih dahulu sebelum bisa menggunakan aplikasi ANTRI.

- b. View Profile
Setiap user dapat melihat informasi dari user yang bersangkutan.
- c. Update Profile
Setiap user dapat mengubah informasi akun user yang bersangkutan.
- d. Memilih Jadwal Dokter
Setiap pasien dapat memilih dokter yang diinginkan dan mulai mengantri.
- e. Ambil Antrian
Setiap pasien dapat mengambil antrian pada dokter yang telah dipilih.
- f. Melihat Sisa Waktu Antrian
Setiap Pasien dapat melihat sisa waktu antrian menuju antriannya.

2.3. Karakteristik dan Klasifikasi Pengguna

Ada tiga tipe user yang berinteraksi dengan sistem: pasien, asisten dokter, dan apoteker. Setiap user memiliki peran dan fungsi berbeda menurut kebutuhan masing-masing jenis user.

Pasien dapat melakukan *login*, melakukan fungsi memilih jadwal dokter untuk memilih dokter dan mulai mengantri. Apabila ingin mengambil nomer antrian untuk poli lain bisa memilih tombol ambil antrian lain. Pasien dapat melihat maupun mengubah informasi akunya.

Asisten dokter bisa langsung melakukan *login*. Kemudian asisten dokter memulai antrian dengan cara menekan tombol yang telah disediakan. Setelah melakukan pemeriksaan atau setelah pasien meninggalkan ruang pemeriksaan, asisten dokter melakukan konfirmasi bahwa pemeriksaan selesai dan

melanjutkan antrian. Asisten dokter bisa juga mengubah jadwal dokter praktek dan memonitor antrian.

Administrator bisa langsung melakukan *login*. Administrator dapat melakukan update data pasien, data dokter, jadwal dokter praktek, menambahkan user baru dan menghapus data dokter.

2.4. Batasan Desain dan Implementasi

- GUI dalam Bahasa Indonesia.
- Login dan password digunakan untuk identifikasi user.
- Hanya pasien, admin dan asisten dokter yang telah terdaftar yang dapat menggunakan layanan ini.

2.5. Dokumentasi Bagi Pengguna

Dokumentasi untuk pengguna yaitu *User Manual* yang akan membantu pengguna memahami aplikasi ANTRI

2.6. Asumsi dan Ketergantungan

- a. Tool pengembangan telah tersedia, dalam hal ini yang dimaksud dengan tool adalah software dan hardware yang dibutuhkan untuk pengembangan.
- b. Semua pasien memiliki smartphone android dan menggunakan aplikasi ANTRI.

3. Kebutuhan Antarmuka Eksternal

3.1. Antarmuka Pengguna User Interface untuk pasien



Gambar 27 UI Home



Gambar 26 UI Login
Pasien



Gambar 29 UI Antrian



Gambar 28 UI Ambil
Antrian

Spesifikasi Kebutuhan Perangkat Lunak



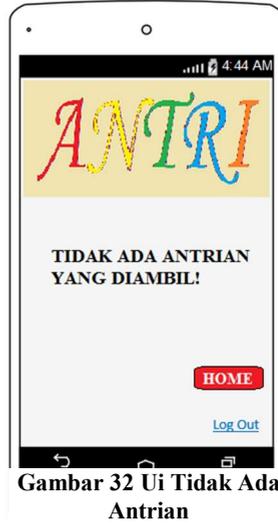
Gambar 30 UI Antrian



Gambar 31 UI Waktu



Gambar 33 UI Ada



Gambar 32 Ui Tidak Ada Antrian

User Interface untuk Administrator

The screenshot shows the login page for the ANTRI system. At the top, the word "ANTRI" is displayed in a colorful, stylized font. Below it, there is a red "Log in" button. The main area contains two input fields: "USERNAME" and "PASSWORD". Below these fields, there are two radio buttons: "Administrasi" (selected) and "Asisten Dokter". At the bottom, there is a "Log In" button.

Gambar 34 UI Login Admin dan AsDok

The screenshot shows the dashboard for an administrator. At the top, the word "ANTRI" is displayed in a colorful, stylized font. Below it, there is a "Selamat Datang, Admin !" message. On the right side, there is a search bar labeled "Cari Data" with a magnifying glass icon. Below the search bar, there is a list of menu items: "Lihat Data Pengguna", "Data Pasien", "Data Dokter", "Jadwal Dokter", and "Lihat Antrian". At the bottom right, there is a "Log Out" link.

Gambar 35 UI Halaman Awal Admin

Spesifikasi Kebutuhan Perangkat Lunak



Gambar 37 Daftar Data Pengguna



Gambar 36 UI Data Pasien

ANTRI

Selamat Datang, Admin !

Tambah Data Pasien

No. Reg 129
 Nama
 Alamat
 TTL
 No. HP
 Alamat Email
 Jenis Kelamin

[Log Out](#)

Cari Data

[Lihat Data Pengguna](#)
[Data Pasien](#)
[Data Dokter](#)
[Jadwal Dokter](#)
[Lihat Antrian](#)

Gambar 38 UI Input Data Pasien

ANTRI

Selamat Datang, Admin !

Edit Data Pasien

No. Reg 123
 Nama An. Kiki
 Alamat Surabaya
 TTL Surabaya, 21-10-2016
 No. HP 465834606506
 Alamat Email kikikiki@kiki
 Jenis Kelamin Perempuan

[Log Out](#)

Cari Data

[Lihat Data Pengguna](#)
[Data Pasien](#)
[Data Dokter](#)
[Jadwal Dokter](#)
[Lihat Antrian](#)

Gambar 39 UI Edit Data Pasien

Spesifikasi Kebutuhan Perangkat Lunak

Selamat Datang, Admin !

Cari Data

Data Dokter

+ Data Dokter

NIK	NAMA	Edit	Hapus
34665	Lisa	Edit	X Hapus
354757	Cici	Edit	X Hapus
353433	Budi	Edit	X Hapus

Lihat Data Pengguna

Data Pasien

Data Dokter

Jadwal Dokter

Lihat Antrian

Log Out

Gambar 40 UI Data Dokter

Selamat Datang, Admin !

Cari Data

Input Data Dokter

NIK 123425487

Nama Siti

Alamat Surabaya

TTL Jakarta, 21-10-1980

No. HP 465834606506

Spesialis Anak

Alamat Email siti@aja

Jenis Kelamin Perempuan

Save Cancel

Lihat Data Pengguna

Data Pasien

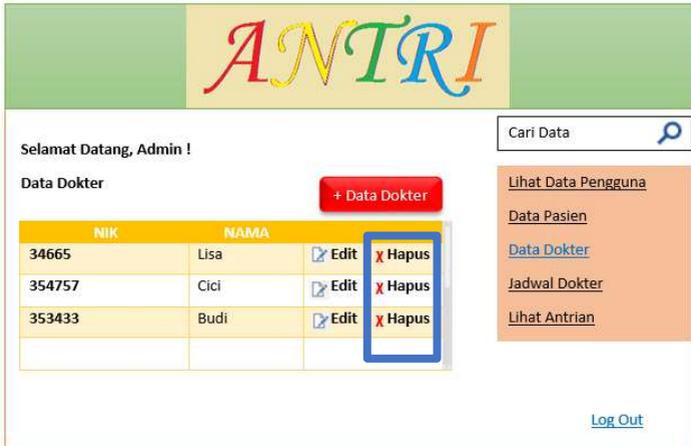
Data Dokter

Jadwal Dokter

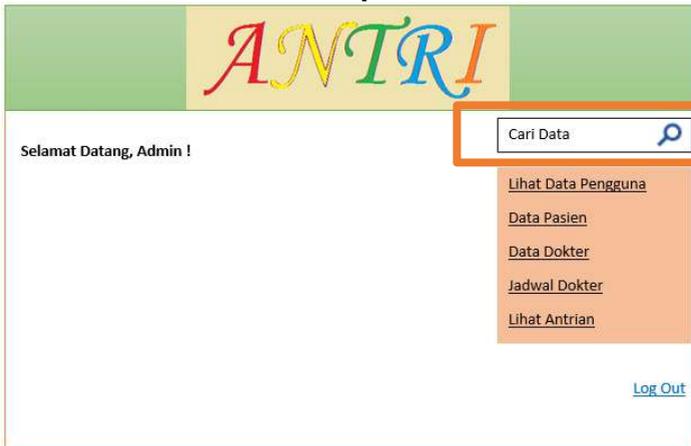
Lihat Antrian

Log Out

Gambar 41 UI Edit Data Dokter



Gambar 42 UI Hapus Data Dokter



Gambar 43 UI Cari Data

Spesifikasi Kebutuhan Perangkat Lunak

Selamat Datang, Admin !

Cari Data

Jadwal Dokter + Tambah Jadwal Dokter

NIK	Nama	Spesialis	Jadwal	Kuota Antrian	
123435	Dr. Arif	Anak	Hari : Senin Pukul : 09.00 – 10.00	20	<input type="checkbox"/> X
124546	Dr. Budi	Kandungan	Hari : Senin Pukul : 09.00 – 10.00		<input type="checkbox"/> X
234356	Dr. Supri	Kandungan	Hari : Senin Pukul : 09.00 – 10.00		<input type="checkbox"/> X

Lihat Data Pengguna
Data Pasien
Data Dokter
Jadwal Dokter
Lihat Antrian

Log Out

Gambar 44 UI Jadwal Dokter

Selamat Datang, Admin !

Cari Data

Tambah Jadwal Dokter

NIK
Nama
Spesialis
Hari/Pukul
Kuota Antrian

Save Cancel

Lihat Data Pengguna
Data Pasien
Data Dokter
Jadwal Dokter
Lihat Antrian

Log Out

Gambar 45 UI Tambah Jadwal Dokter

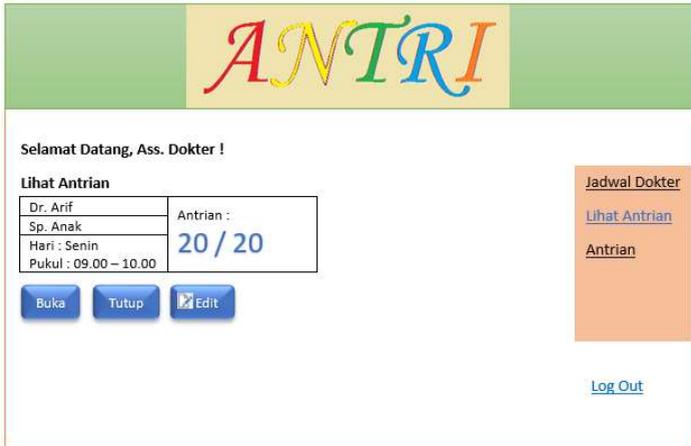


Gambar 46 UI Ubah Jadwal Dokter

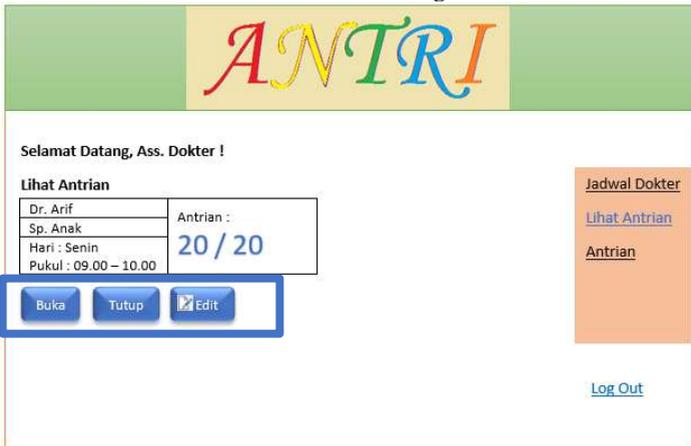


Gambar 47 UI Menonaktifkan Jadwal Dokter

User Interface untuk Asisten Dokter



Gambar 48 UI Monitoring Antrian



Gambar 49 UI Set Antrian



Gambar 50 UI Next Antrian



Gambar 51 UI Panggil Antrian

ANTRI

Selamat Datang, Admin ! Cari Data

Lihat Antrian + Tambah Antrian Baru

NIK	Nama	Sp.	Jadwal	Antrian
123435	Dr. Arif	Anak	Hari : Senin Pukul : 09.00 – 10.00	20/20 Edit x nonaktifkan

Lihat Data Pengguna

[Data Pasien](#)

[Data Dokter](#)

[Jadwal Dokter](#)

[Lihat Antrian](#)

[Log Out](#)

Gambar 52 UI Lihat Antrian Admin

ANTRI

Selamat Datang, Ass. Dokter !

Lihat Antrian

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Dr. Arif</td></tr> <tr><td>Sp. Anak</td></tr> <tr><td>Hari : Senin</td></tr> <tr><td>Pukul : 09.00 – 10.00</td></tr> </table>	Dr. Arif	Sp. Anak	Hari : Senin	Pukul : 09.00 – 10.00	<p>Antrian : 20 / 20</p> <div style="display: flex; justify-content: space-around; margin-top: 5px;"> Buka Tutup Edit </div>
Dr. Arif					
Sp. Anak					
Hari : Senin					
Pukul : 09.00 – 10.00					

[Jadwal Dokter](#)

[Lihat Antrian](#)

[Antrian](#)

[Log Out](#)

Gambar 53 UI Lihat Antrian AssDok

3.2. Antarmuka Perangkat Keras

Perangkat keras yang dibutuhkan pada pembuatan aplikasi ini adalah laptop *HP* dengan spesifikasi sebagai berikut :

1. *Processor* Intel Core i3-5005U 2.00 GHz
2. RAM 4GB DDR3 PC12800 max 15GB
3. *Harddisk* 750GB + 128GB SSD
4. Windows 10 64-bit

Implementasi pada perangkat android menggunakan Asus ZE551ML dengan spesifikasi sebagai berikut :

1. *Processor* Quad-core 2.3 GHz
2. *OS* Android OS, Android Lollipop (v5.0)
3. *Memory Internal* 68 GB, 2 GB RAM

3.3. Antarmuka Perangkat Lunak

- Client Side
OS Android Lollipop (5.0)
- Petugas
OS Windows
- Asisten Dokter
OS Windows
- Development End
PowerDesign, HTML, PHP, Database MySQL, OS (Windows).

3.4. Antarmuka Komunikasi

Sistem ANTRI akan menggunakan HTTP protocol untuk komunikasi yang dilakukan melalui internet sedangkan untuk komunikasi yang dilakukan melalui intranet akan menggunakan rangkaian TCP/IP protocol.

4. Feature Sistem

Kebutuhan fungsional :

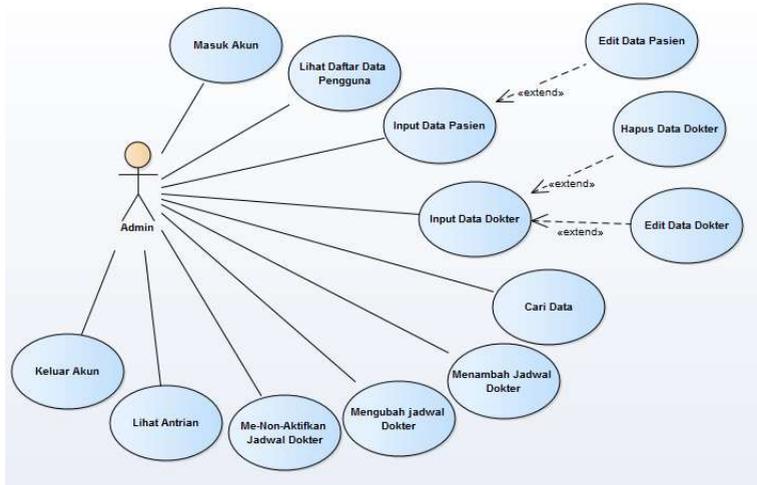
Tabel 12 Kebutuhan Fungsional

KF-01	Sistem menyediakan fitur bagi pengguna untuk <i>login</i> dan <i>logout</i>
KF-02	Sistem menyediakan fitur bagi admin untuk mengelola data pengguna (pasien) dan data dokter
KF-03	Sistem menyediakan fitur bagi admin untuk mengelola jadwal dokter (jam, hari, batas maksimal kesanggupan dokter)(default)
KF-04	Sistem menyediakan fitur bagi asisten dokter(perawat) untuk melakukan update antrian
KF-05	Sistem menyediakan fitur untuk menampilkan daftar antrian dan sisa antrian
KF-06	Sistem menyediakan fitur bagi pasien untuk pengambilan antrian

Use-Case Diagram

UC Administrator

Spesifikasi Kebutuhan Perangkat Lunak



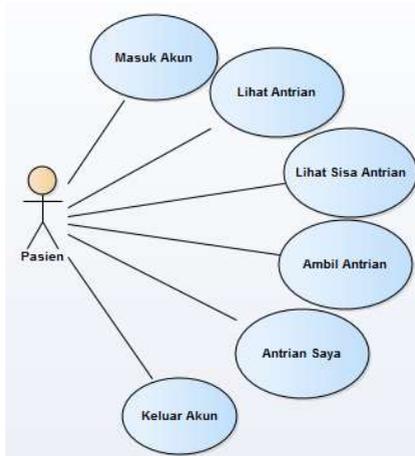
Gambar 54 UCD Admin

UC Asisten Dokter



Gambar 55 UCD AssDok

UC Pasien



Gambar 56 UCD Pasien

Use Case

Tabel 13 Use-case

	Kode use-case	Use-case	Aktor
KF-01	UC-01.01	Masuk akun	Semua Pengguna
	UC-01.02	Keluar akun	Semua Pengguna
KF-02	UC-02.01	Lihat Daftar Data Pengguna	Administrator
	UC-02.02	Input Data Dokter	Administrator
	UC-02.03	Input Data Pasien	Administrator
	UC-02.04	Edit Data Dokter	Administrator
	UC-02.05	Edit Data Pasien	Administrator
	UC-02.06	Hapus Data Dokter	Administrator
	UC-02.07	Cari Data	Administrator
KF-03	UC-03.01	Menambah jadwal dokter	Administrator
	UC-03.02	Mengubah jadwal dokter	Administrator dan asisten dokter (perawat)
	UC-03.03	Me-non-aktifkan jadwal dokter	Administrator
KF-04	UC-04.01	Monitoring Antrian	Asisten Dokter (perawat)
	UC-04.02	Set Antrian	Asisten Dokter (perawat)
	UC-04.03	Panggil Antrian	Asisten Dokter
	UC-04.04	Next Antrian	Asisten Dokter
KF-05	UC-05.01	Lihat Antrian	Semua Pengguna
KF-06	UC-06.01	Ambil Antrian	Pasien
	UC-06.02	Antrian Saya	Pasien

Use-case berdasarkan area fungsional

Pengelolaan Akun

Tabel 14 Use-Case Pengelolaan Akun

Kode use-case	Use-case
UC-01.01	Masuk akun
UC-01.02	Keluar akun
UC-02.01	Lihat Daftar Data Pengguna
UC-02.02	Input Data Dokter
UC-02.03	Input Data Pasien
UC-02.04	Edit Data Dokter
UC-02.05	Edit Data Pasien
UC-02.06	Hapus Data Dokter
UC-02.07	Cari Data

Pengelolaan Antrian

Tabel 15 Use-Case Pengelolaan Antrian

Kode use-case	Use-case
UC-04.01	Monitoring Antrian
UC-04.02	Set Antrian
UC-04.03	Panggil Antrian
UC-04.04	Next Antrian
UC-05.01	Lihat Antrian
UC-06.01	Ambil Antrian
UC-06.02	Antrian Saya

Pengelolaan Jadwal

Tabel 16 Use-Case Pengelolaan Jadwal

Kode use-case	Use-case
UC-03.01	Menambah jadwal dokter
UC-03.02	Mengubah jadwal dokter
UC-03.03	Me-non-aktifkan jadwal dokter

Use-case berdasarkan pengguna

Administrator

Tabel 17 Use-Case Use-Case Administrator

Kode use-case	Use-case
UC-01.01	Masuk akun
UC-01.02	Keluar akun
UC-02.01	Lihat Daftar Data Pengguna
UC-02.02	Input Data Dokter
UC-02.03	Input Data Pasien
UC-02.04	Edit Data Dokter
UC-02.05	Edit Data Pasien
UC-02.06	Hapus Data Dokter
UC-02.07	Cari Data
UC-03.01	Menambah jadwal dokter
UC-03.02	Mengubah jadwal dokter
UC-03.03	Me-non-aktifkan jadwal dokter

Asisten Dokter (Perawat)

Tabel 18 Use-Case Asisten Dokter

Kode use-case	Use-case
UC-01.01	Masuk akun
UC-01.02	Keluar akun
UC-03.02	Mengubah jadwal dokter
UC-04.01	Monitoring Antrian
UC-04.02	Set Antrian
UC-04.03	Panggil Antrian
UC.04.04	Next Antrian

Pasien

Tabel 19 Use-Case Pasien

Kode use-case	Use-case
UC-01.01	Masuk akun
UC-01.02	Keluar akun
UC-05.01	Lihat Antrian
UC-06.01	Ambil Antrian
UC-06.02	Antrian Saya

Use-case deskripsi

Tabel 20 UCD Masuk Akun(Login)

+	UC-01.01 Masuk Akun	
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar aktor dapat masuk ke dalam sistem antrian dengan masing-masing wewenang.	
Actor :	Semua Pengguna	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Sekali saja setiap user (<i>once</i>)	
Pre Condition :	Aktor telah masuk halaman login	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memasukkan “<i>Username</i>” dan “<i>Password</i>” pada <i>form</i> masuk. 2. Aktor menekan tombol “<i>Login</i>” untuk masuk ke dalam sistem antrian. 3. Sistem akan memeriksa apakah <i>form</i> “<i>Username</i>” dan “<i>Password</i>” telah sesuai dengan <i>database</i> jika tidak sesuai maka akan dijalankan skenario alternatif I no 1 	
Skenario Alternatif:	<ul style="list-style-type: none"> ❖ Alternatif I <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan dibawah kolom nama akun “Maaf <i>username</i> dan <i>password</i> tidak sesuai, mohon dicoba lagi” 	

Tabel 21 UCD Keluar Akun

+	UC-01.02 Keluar Akun	
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar aktor dapat keluar dari sistem antrian	
Direct Actor :	Semua Pengguna	

Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Sekali saja setiap user (<i>once</i>)
Pre Condition :	Aktor sebelumnya telah melakukan “ <i>Login</i> ”
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih menu “<i>Logout</i>”. 2. Sistem akan kembali ke halaman utama sistem antrian.
Skenario Alternatif:	-

Tabel 22 UCD Lihat Daftar Data Pengguna

+	UC-02.01 Lihat daftar data pengguna
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk melihat daftar data pengguna
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Kelola Data” 2. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 3. Aktor memilih salah satu menu (misal) “Data Pasien” 4. Sistem akan menampilkan halaman “List Data Pasien”

Skenario Alternatif:	-
---------------------------------	---

Tabel 23 UCD Input Data Dokter

+	UC-02.02 Input Data Dokter
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk menambah data dokter
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Kelola Data” 2. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 3. Aktor memilih menu “Data Dokter” 4. Sistem akan menampilkan halaman “List Data Dokter” 5. Aktor menekan icon tambah pada halaman tersebut 6. Sistem akan menampilkan halaman “Tambah Data Dokter” 7. Aktor mengisi data dokter “NIK”, “Nama”, “Tempat Tanggal Lahir”, “Alamat”, “Nomer Hp”, “Spesialis”, dan “Alamat Email” <ol style="list-style-type: none"> c. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 d. Apabila <i>field</i> “NIK” sudah terpakai, maka sistem akan menjalankan alternatif I no. 2 8. Aktor menekan tombol

Skenario Alternatif:	<ul style="list-style-type: none"> c. “Save”, maka sistem akan menampilkan halaman pemberitahuan “Penambahan Data Berhasil” d. “Cancel”, maka sistem akan menampilkan halaman “List Data Dokter” <p>9. Sistem akan memperbarui Data Dokter dan kembali ke halaman “List Data Dokter”</p>
	<ul style="list-style-type: none"> ❖ Alternatif I <ul style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 2. Sistem akan menampilkan peringatan (<i>notification</i>), yang berisi bahwa “NIK sudah terpakai/terdaftar”

Tabel 24 UCD Input Data Pasien

+	UC-02.03 Input Data Pasien
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk menambah data pasien
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Kelola Data” 2. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 3. Aktor memilih menu “Data Pasien” 4. Sistem akan menampilkan halaman “List Data Pasien” 5. Aktor menekan icon tambah pada halaman tersebut

Skenario Alternatif:	<ol style="list-style-type: none"> 6. Sistem akan menampilkan halaman “Tambah Data Pasien” 7. Aktor mengisi data dokter “No. Regristasi”, “Nama”, “Tempat Tanggal Lahir”, “Alamat”, “Nomer Hp” dan “Alamat Email” <ol style="list-style-type: none"> e. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 f. Apabila <i>field</i> “No. Hp” sudah terpakai, maka sistem akan menjalankan alternatif I no. 2 8. Aktor menekan tombol <ol style="list-style-type: none"> e. “Save”, maka sistem akan menampilkan halaman pemberitahuan “Penambahan Data Berhasil” f. “Cancel”, maka sistem akan menampilkan halaman “List Data Pasien” 9. Sistem akan memperbarui Data Pasien dan kembali ke halaman “List Data Pasien”
	<p>❖ Alternatif I</p> <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 2. Sistem akan menampilkan peringatan (<i>notification</i>), yang berisi bahwa “No. Hp sudah terpakai/terdaftar”

Tabel 25 UCD Edit Data Dokter

+	UC-02.04 Edit Data Dokter
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengubah data dokter
Direct Actor :	Administrator

Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 8. Aktor memilih tombol menu “Kelola Data” 9. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 10. Aktor memilih menu “Data Dokter” 11. Sistem akan menampilkan halaman “List Data Dokter” 12. Aktor menekan icon edit () pada salah satu kolom dokter yang ingin di edit 13. Sistem akan menampilkan halaman “Data Dokter ‘A’” 14. Aktor mengisi ulang data dokter yang ingin diganti “NIK”, “Nama”, “Tempat Tanggal Lahir”, “Alamat”, “Nomer Hp”, “Spesialis”, dan “Alamat Email” <ol style="list-style-type: none"> g. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 h. Apabila <i>field</i> “NIK” sudah terpakai, maka sistem akan menjalankan alternatif I no. 2 15. Aktor menekan tombol <ol style="list-style-type: none"> g. “Save”, maka sistem akan menampilkan halaman pemberitahuan “Penambahan Data Berhasil” h. “Cancel”, maka sistem akan menampilkan halaman “List Data Dokter”

Skenario Alternatif:	16. Sistem akan memperbarui Data Dokter dan kembali ke halaman “List Data Dokter”
	❖ Alternatif I 3. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 4. Sistem akan menampilkan peringatan (<i>notification</i>), yang berisi bahwa “NIK sudah terpakai”

Tabel 26 UCD Edit Data Pasien

+	UC-02.05 Edit Data Pasien
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengubah data pasien
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Kelola Data” 2. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 3. Aktor memilih menu “Data Pasien” 4. Sistem akan menampilkan halaman “List Data Pasien” 5. Aktor menekan icon edit () pada salah satu kolom pasien yang ingin di edit 6. Sistem akan menampilkan halaman “Data Pasien ‘C’ ”

	<ol style="list-style-type: none"> 7. Aktor mengisi ulang data dokter yang ingin diganti “No. Registrasi”, “Nama”, “Tempat Tanggal Lahir”, “Alamat”, “Nomer Hp” dan “Alamat Email” <ol style="list-style-type: none"> a. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 b. Apabila <i>field</i> “No. Hp” sudah terpakai, maka sistem akan menjalankan alternatif I no. 2 8. Aktor menekan tombol <ol style="list-style-type: none"> a. “Save”, maka sistem akan menampilkan halaman pemberitahuan “Penambahan Data Berhasil” b. “Cancel”, maka sistem akan menampilkan halaman “List Data Pasien” 9. Sistem akan memperbarui Data Pasien dan kembali ke halaman “List Data Pasien”
<p>Skenario Alternatif:</p>	<p>❖ Alternatif I</p> <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 2. Sistem akan menampilkan peringatan (<i>notification</i>), yang berisi bahwa “No. Hp sudah terpakai/terdaftar”

Tabel 27 UCD Hapus Data Dokter

+	UC-02.06 Hapus Data Dokter
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk menghapus data dokter
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih tombol menu “Kelola Data” 2. Sistem menampilkan pilihan “Data Pasien” dan “Data Dokter” 3. Aktor memilih menu “Data Dokter” 4. Sistem akan menampilkan halaman “List Data Dokter” 5. Aktor menekan icon edit (χ) pada salah satu kolom dokter yang ingin di hapus 6. Sistem akan menampilkan pertanyaan “Apakah Yakin akan menghapus dokter ‘A’? “ 7. Aktor menekan tombol <ol style="list-style-type: none"> a. “Ya”, maka sistem akan menghapus data dokter tersebut dan menampilkan halaman pemberitahuan “Hapus Data Berhasil” b. “Cancel”, maka sistem akan menampilkan halaman “List Data Dokter”

Skenario Alternatif:	8. Sistem akan menampilkan List Data Dokter terbaru
	❖ Alternatif I -

Tabel 28 UCD Cari Data

+	UC-02.07 Cari Data
Ringkasan :	Use-case ini dimaksudkan agar admin dapat mencari user yang telah terdaftar pada aplikasi
Direct Actor :	Administrator
Prioritas :	Pilihan (<i>Optional</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<p>Untuk melakukan pencarian user dapat dilakukan dengan cara, yaitu :</p> <ol style="list-style-type: none"> 1. Aktor memasukkan kata kunci berupa nama dari pengguna pada kolom pencarian. 2. Sistem menampilkan hasil pencarian berdasarkan kata kunci yang dimasukkan, dari semua peran akun.(dokter maupun pasien)
Skenario Alternatif:	-

Tabel 29 UCD Tambah Jadwal Dokter

+	UC-03.01 Menambah jadwal dokter	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk Menambah Jadwal Dokter	
Direct Actor :	Administrator	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)	
Pre Condition :	Aktor telah masuk di halaman administrator	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih menu “Jadwal Dokter” 2. Sistem menampilkan Halaman List Jadwal Dokter 3. Aktor menekan tombol “Tambah Jadwal Dokter” pada salah satu kolom dokter 4. Sistem Menampilkan Form “Tambah Jadwal Dokter” 5. Aktor mengisi field “NIK”, “Jam”, “Hari”, “Kuota pasien” <ol style="list-style-type: none"> a. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 6. Aktor menekan tombol <ol style="list-style-type: none"> a. “Save”. Maka sistem akan menampilkan halaman “List Jadwal Dokter” terbaru b. “Cancel”, maka sistem akan kembali ke halaman “List Jadwal Dokter” 	
Skenario Alternatif:	❖ Alternatif I <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 	

Tabel 30 UCD Ubah Jadwal Dokter

+	UC-03.02 Mengubah jadwal dokter	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengubah jadwal dokter	
Direct Actor :	Administrator dan Asisten Dokter(perawat)	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)	
Pre Condition :	Aktor telah masuk di halaman administrator	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih menu “Jadwal Dokter” 2. Sistem menampilkan Halaman List Jadwal Dokter 3. Aktor menekan tombol “Ubah Jadwal Dokter” pada salah satu kolom dokter 4. Sistem Menampilkan Form “Ubah Jadwal Dokter” 5. Aktor mengisi field “NIK”, “Jam”, “Hari”, “Kuota pasien” <ol style="list-style-type: none"> a. Apabila terdapat <i>field</i> yang kosong, maka sistem akan menjalankan alternatif I. no 1 6. Aktor menekan tombol <ol style="list-style-type: none"> a. “Save”. Maka sistem akan menampilkan halaman “List Jadwal Dokter” terbaru b. “Cancel”, maka sistem akan kembali ke halaman “List Jadwal Dokter” 	
Skenario Alternatif:	❖ Alternatif I <ol style="list-style-type: none"> 1. Sistem akan menampilkan pesan pesan bahwa “Kolom harus diisi” 	

Tabel 31 UCD Menonaktifkan Jadwal Dokter

+	UC-03.03 Me-Non-Aktifkan Jadwal Dokter
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk men-non-aktifkan jadwal praktek dokter(Disable)
Direct Actor :	Administrator
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Jarang (<i>rarely</i>)
Pre Condition :	Aktor telah masuk di halaman administrator
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor memilih menu “Jadwal Dokter” 2. Sistem menampilkan Halaman List Jadwal Dokter 3. Aktor memilih salah satu kolom data dokter yang akan di disable 4. Sistem menampilkan data dokter ‘A’ 5. Aktor mencentang field disable pada bagian bawah halaman data dokter 6. Sistem menampilkan pertanyaan “Apakah Yakin Akan Men-Non-Aktifkan Dokter ‘A’? ” 7. Aktor menekan tombol <ol style="list-style-type: none"> a. “Ya”. Maka sistem akan menampilkan halaman “List Jadwal Dokter” terbaru b.“Cancel”, maka sistem akan kembali ke halaman “List Jadwal Dokter”
Skenario Alternatif:	-

Tabel 32 UCD Monitoring Antrian

+	UC-04.01 Monitoring Antrian	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk membuka antrian pada waktu dokter praktek	
Direct Actor :	Asisten Dokter(perawat)	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Sering	
Pre Condition :	Aktor telah masuk di halaman Asisten Dokter	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Lihat Antrian” 2. Sistem akan menampilkan pilihan “Daftar Antrian” 	
Skenario Alternatif:	-	

Tabel 33 UCD Set Antrian

+	UC-04.02 Set Antrian	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengupdate antrian	
Direct Actor :	Asisten Dokter	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Sering	
Pre Condition :	Aktor telah masuk di halaman asisten dokter	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Lihat Antrian” 2. Sistem akan menampilkan pilihan “Daftar Antrian” 	

Skenario Alternatif:	<ol style="list-style-type: none"> 3. Aktor memilih salah satu tombol “Buka”, “Tutup” dan “Edit” pada masing-masing dokter yang praktek” 4. Sistem akan mengupdate antrian
	-

Tabel 34 UCD Panggil Antrian

+	UC-04.03 Panggil Antrian
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk memanggil nomer antrian
Direct Actor :	Asisten Dokter
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Sering
Pre Condition :	Aktor telah masuk di halaman asisten dokter
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Antrian” 2. Sistem akan menampilkan pilihan “Daftar Antrian” 3. Aktor menekan tombol “Panggil Antrian” 4. Sistem akan memanggil atau menyebutkan nomer antrian melalui speaker
Skenario Alternatif:	-
Exception:	<ol style="list-style-type: none"> 1. AssDok sudah menekan tombol Panggil Tiga kali,tetapi pasien tidak datang, maka AssDok

	<p>menginformasikan kepada administrasi.</p> <ol style="list-style-type: none">2. Pihak Administrasi mencoba menghubungi pasien3. Pasien Datang terlambat, pasien menunggu hingga akhir antrian
--	--

Tabel 35 UCD Next Antrian

+	UC-04.03 Next Antrian	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengupdate antrian	
Direct Actor :	Asisten Dokter	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Sering	
Pre Condition :	Aktor telah masuk di halaman asisten dokter	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Antrian” 2. Sistem akan menampilkan pilihan “Daftar Antrian” 3. Aktor menekan tombol “Next Antrian” 4. Sistem akan mengupdate antrian 	
Skenario Alternatif:	-	

Tabel 36 UCD Lihat Antrian

+	UC-05.01 Lihat Antrian	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk melihat Antrian	
Direct Actor :	Pasien	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Jarang	
Pre Condition :	Aktor telah masuk di halaman pasien	

Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Antrian” 2. Sistem akan menampilkan pilihan “List Daftar Dokter Praktek” 3. Aktor menekan tombol “Lihat Antrian” pada masing-masing dokter yang praktek” 4. Sistem akan menampilkan Antrian pada dokter ‘A’
Skenario Alternatif:	-

Tabel 37 UCD Ambil Antrian

+	UC-06.01 Ambil Antrian
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk mengambil nomer antrian bagi pasien
Direct Actor :	Pasien
Prioritas :	Penting (<i>essential</i>)
Frekuensi Penggunaan :	Sering
Pre Condition :	Aktor telah masuk di halaman Pasien
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Sistem menampilkan halaman awal untuk pasien, yang terdapat tombol “Antrian Dokter” 2. Aktor menekan menu “Antrian Dokter” 3. Sistem akan menampilkan pilihan “List Daftar Dokter Praktek” 4. Aktor menekan tombol “Lihat Antrian” pada masing-masing dokter yang praktek” 5. Sistem akan menampilkan Antrian pada dokter ‘A’ 6. Aktor menekan tombol “Ambil Antrian” <ol style="list-style-type: none"> a. Apabila Kuota Tersedia, maka sistem akan menjalankan alternatif I no. 1

Skenario Alternatif:	b. Apabila Kuota Full, maka sistem akan menjalankan alternatif I no. 2
	<ul style="list-style-type: none">❖ Alternatif I1. Sistem akan menampilkan nomer Antrian yang didapat pasien2. Sistem akan menampilkan pemberitahuan “Kuota Antrian Full, Silahkan hubungi Administrasi Rumah sakit di No. 0318888888”

Tabel 38 UCD Antrian Saya

+	UC-06.02 Antrian Saya	
Ringkasan :	<i>Use-case</i> ini dimaksudkan untuk melihat waktu sisa antrian	
Direct Actor :	pasien	
Prioritas :	Penting (<i>essential</i>)	
Frekuensi Penggunaan :	Sering (<i>rarely</i>)	
Pre Condition :	Aktor telah masuk di halaman utama sebagai pasien	
Skenario Sukses Utama :	<ol style="list-style-type: none"> 1. Aktor menekan menu “Antrian Saya” 2. Sistem akan menampilkan halaman “Antrian Saya” 3. Apabila terdapat Antrian yang di ambil maka sistem akan menjalankan alternatif I no. 1 4. Apabila tidak ada Anrtrian yang di ambil maka sistem akan menjalankan alternatif I no. 2 	
Skenario Alternatif:	<ul style="list-style-type: none"> ❖ Alternatif I <ol style="list-style-type: none"> 1. Sistem akan menampilkan Nomer Antrian beserta waktu antrian 2. Sistem akan menampilkan notifikasi “Tidak Ada Antrian yang di ambil” 	

5. Kebutuhan NonFungsional Lainnya

5.1. Usability requirement

Kemudahan Penggunaan

Tabel 39 KNF Usability

KNF-01	Semua fitur yang tersedia dapat digunakan sebagaimana fungsinya
KNF-02	Tidak ada menu atau tombol yang membingungkan bagi pengguna

--	--

5.2. Reliability and security requirement

Kehandalan Sistem Dan Keamanan

Tabel 40 KNF Reliability and Security

KNF-03	Sistem dapat menampilkan seluruh konten dari sistem antrian
KNF-04	Hanya admin yang dapat mendaftarkan atau menambah pengguna
KNF-05	Hanya admin dan asisten dokter yang dapat mengupdate antrian

5.3. Portability Requirement

Kemudahan Dalam Pengaksesan

Tabel 41 KNF Portability

KNF-06	Waktu untuk penanganan pada saat server down maksimal 1x24 jam (1 hari)
--------	---

5.4. Supportability requirement

Dukungan Dalam Penggunaan Sistem

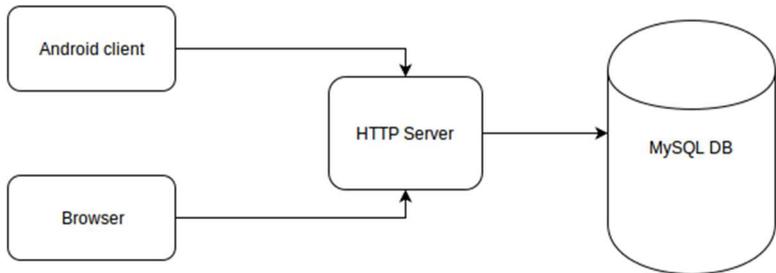
Tabel 42 KNF Supportability

KNF-07	Sistem dapat diakses di segala OS(operating system) untuk website
KNF-08	Sistem dapat diakses di segala OS Android untuk mobile

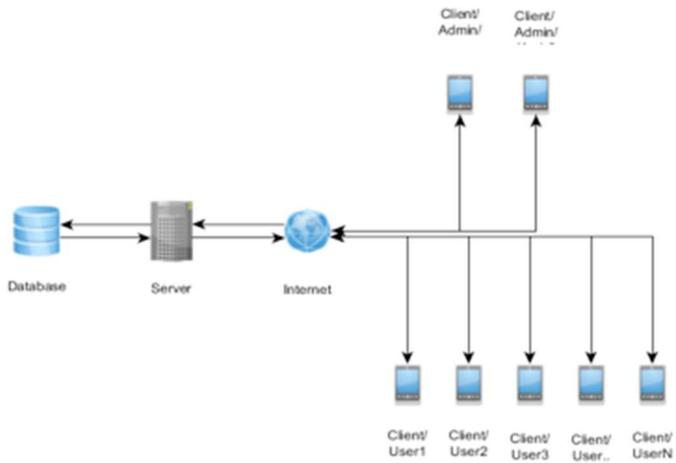
6. Kebutuhan Lain

Aplikasi ANTRI memerlukan lisensi atau hak paten atas kepemilikan software ini.

Appendix A: Model Analisis



Gambar 57 Arsitektur Sistem



Gambar 58 Arsitektur Sistem lain

Appendix B: Daftar Kebutuhan

Algoritma
Autorisasi
Change
Constraint
Database
Functional Requirements
Hardware Interface
Login
Registration
Requirement
Non-Functional Requirements
Select
Software
Software Interface
Update
User Interface
View

LAMPIRAN D – DDPL**DDPL-01****DOKUMEN DESAIN PERANGKAT LUNAK**

Dipersiapkan oleh:

Nurul Fakhria 5209100097

sebagai kelengkapan Tugas Akhir di Jurusan Sistem Informasi
ITS

DOKUMEN DESAIN PERANGKAT LUNAK



Dipersiapkan oleh:

Nurul Fakhria 5209100097

sebagai kelengkapan Tugas Akhir di Jurusan Sistem Informasi ITS

Daftar Perubahan

Tabel 43 Daftar Perubahan

Revisi	Deskripsi
A	
B	
C	
D	

Daftar Halaman Perubahan

Tabel 44 Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

Daftar Isi.....	ii
Daftar Tabel.....	iii
Daftar Gambar.....	iv
1 Pendahuluan.....	1
1.1 Tujuan.....	1
1.2 Ruang Lingkup Masalah.....	1
1.3 Definisi dan Istilah.....	1
1.4 Aturan Penamaan dan Penomoran.....	1
1.5 Referensi.....	2
1.6 Rancangan Lingkungan Implementasi.....	2
1.7 Deskripsi Data.....	3
1.7.1 Entity Relationship Data.....	6
1.7.2 Design Constraints.....	8
1.7.3 Logical Database Requirements.....	8
1.7.4 Sequence Diagrams.....	9
1.7.5 Class Diagram.....	18
1.7.5.1 Data Flow Diagrams (DFD).....	18
1.8 Change Management Process.....	18

Daftar Tabel

Tabel 1 Daftar Perubahan.....	i
Tabel 2 Daftar Halaman Perubahan	i
Tabel 3 Penomoran.....	2
Tabel 4 Tabel Pasien	3
Tabel 5 Tabel Dokter.....	4
Tabel 6 Tabel Antrian.....	4
Tabel 7 Tabel Asisten Dokter.....	4
Tabel 8 Tabel Jadwal Dokter.....	5

Daftar Gambar

Gambar 1 ER Diagram.....	3
Gambar 2 ERD.....	6
Gambar 3 Entity Relationship Diagram.....	7
Gambar 4 SD LogIn.....	9
Gambar 5 SD LogOut.....	9
Gambar 6 SD Lihat Daftar Data Pengguna.....	10
Gambar 7 SD Cari Data.....	10
Gambar 8 SD Edit Data Dokter.....	11
Gambar 9 SD Edit Data Pasien.....	11
Gambar 10 SD Hapus Data Dokter.....	12
Gambar 11 SD Input Data Dokter.....	12
Gambar 12 SD Input Data Pasien.....	13
Gambar 13 SD Tambah Jadwal Dokter.....	13
Gambar 14 SD Mengubah Jadwal Dokter.....	14
Gambar 15 SD Menonaktifkan Jadwal Dokter.....	14
Gambar 16 SD Monitoring Antrian.....	15
Gambar 17 SD Set Antrian.....	15
Gambar 18 SD Antrian.....	16
Gambar 19 SD Lihat Antrian.....	17
Gambar 20 SD Ambil Antrian.....	17
Gambar 21 SD Antrian Saya.....	18
Gambar 22 Data Flow Diagram.....	18

Pendahuluan

1.1 Tujuan

Tujuan dari dokumen ini adalah mendeskripsikan implementasi Aplikasi ANTRI yang dijelaskan dalam kebutuhan bisnis rumah sakit. Aplikasi ini dirancang untuk menggantikan sistem manual yang ada.

1.2 Ruang Lingkup Masalah

- Terdapat 3 user, yaitu pasien, administrator (Bagian Administrasi), dan Asisten Dokter.
- Semua user memiliki profil mereka sendiri dalam database
- Pasien dapat mencari jadwal dokter dan mengantri.
- Asisten Dokter dapat membuka dan mengatur antrian.

1.3 Definisi dan Istilah

Guna memberikan gambaran yang sama terhadap beberapa definisi, istilah dan singkatan yang digunakan di dokumen ini, perlu dijelaskan sebagaimana berikut:

IEEE : Institute of Electrical and Electronics Engineers
Adalah standart internasional untuk pengembang perangkat lunak

DDPL : Dokumen Desain Perangkat Lunak
Adalah dokumen hasil perancangan sebuah perangkat lunak yang berisi desain data dan desain tabel yang menunjang pembuatan aplikasi.

1.4 Aturan Penamaan dan Penomoran

Ada beberapa hal/bagian dalam dokumen ini yang perlu diberi nomor. Maksud penomoran ini untuk mempermudah audience dalam pengidentifikasian. Adapun aturan penomorannya sebagaimana tabel berikut:

Tabel 45 Penomoran

Hal/Bagian	Aturan Penomoran
Tabel/Data Store	Nomor berbentuk TD99, dimana 99 adalah nomor urut tabel atau data store Contoh: TD11, TD12, TD29, TD31 dan sebagainya
Sequence Diagram	Contoh: Gambar Sequence 1 dan sebagainya
Class Diagram	Contoh: Class Diagram 1 dan sebagainya

1.5 Referensi

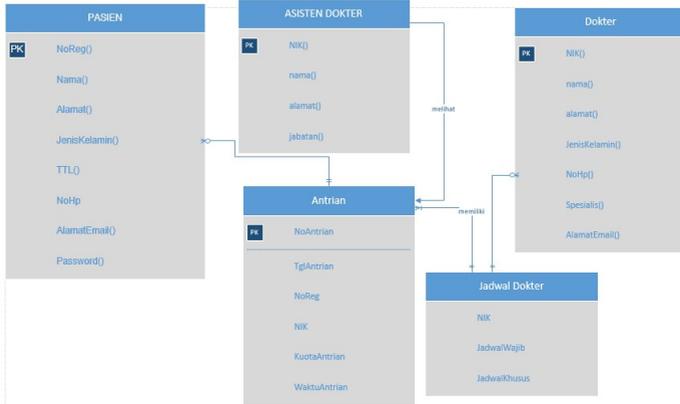
Berikut adalah daftar acuan yang digunakan dalam pendokumentasian spesifikasi kebutuhan perangkat lunak ini.

- IEEE, Software Requirements Engineering, Second Edition, IEEE Computer Society Press, 2002.
- Bray, Ian K. An Introduction to Requirement Engineering, 1st published, Addison-Wesley, 2002
- Kotonya, Gerald and Sommerville, Ian. Requirement Engineering: Processes and Techniques, John Wiley & Sons Ltd, 1998
- Dokumen SKPL
- Wawancara dengan bagian administrasi rumah sakit.

1.6 Rancangan Lingkungan Implementasi

1. System aplikasi android di develop menggunakan Android Studio
2. Komunikasi antara portal aplikasi web dengan database menggunakan SQL
3. Tampilan aplikasi web akan di coding dengan HTML
4. Product akan di coding menggunakan PHP
5. System hanya bisa di run pada Sistem Operasi Windows dan Sistem Operasi Android minimal versi 4.0

1.7 Deskripsi Data



Gambar 59 ER Diagram

Bagian ini akan menjelaskan struktur data, sebagai hasil dari analisis ERD.

Table yang terbentuk ada 5 (Lima) dengan nama masing masing tablenya adalah sebagai berikut:

- Table Pasien
- Table Assisten Dokter
- Table Dokter
- Table Jadwal Dokter
- Table Antrian

Penjelasan fungsi dari masing masing tabel akan dijelaskan pada bagian berikut ini.

Tabel Pasien

digunakan untuk menyimpan informasi Pasien

Tabel 46 Tabel Pasien

Nama Field	Tipe
NoReg	Integer
Nama	Varchar(50)
Alamat	Varchar(250)
Jenis Kelamin	Varchar(10)
TTL	Varchar(10)

No.Hp	Integer
Alamat Email	Varchar(50)
Password	Varchar (10)

Tabel Dokter

digunakan untuk menyimpan informasi Dokter

Tabel 47 Tabel Dokter

Nama Field	Tipe
NIK	Integer
Nama	Varchar(50)
Alamat	Varchar(250)
Jenis Kelamin	Varchar(10)
No.Hp	Integer
Spesialis	Varchar(10)
Alamat Email	Varchar(50)

Tabel Antrian

digunakan untuk menyimpan informasi Antrian

Tabel 48 Tabel Antrian

Nama Field	Tipe
No. Antrian	Integer
Tgl Antrian	Date & Time
NoReg	Integer
NIK	Integer
Kuota Antrian	Varchar(2)
Waktu Antrian	Date & Time

Tabel Asisten Dokter

digunakan untuk menyimpan informasi Asisten Dokter

Tabel 49 Tabel Asisten Dokter

Nama Field	Tipe
NIK	Integer
Nama	Varchar(50)
Alamat	Varchar(250)
Jabatan	Varchar(50)

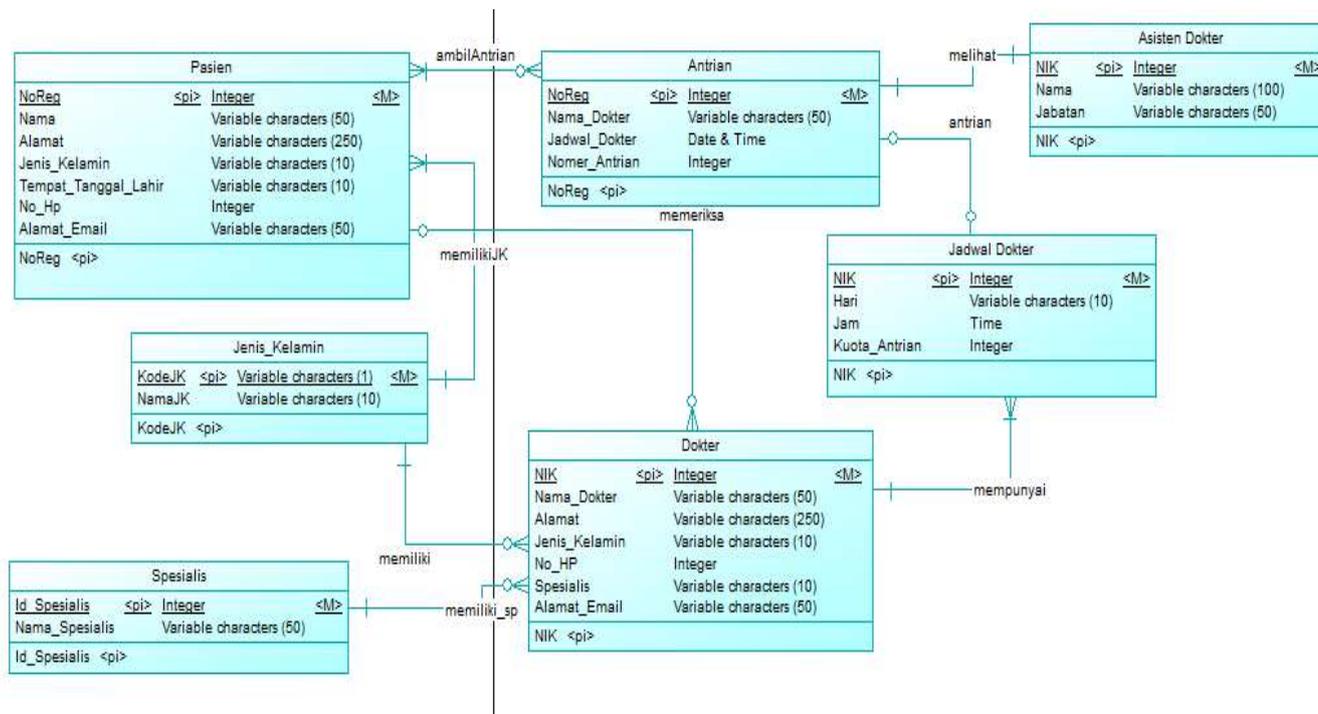
Tabel Jadwal Dokter

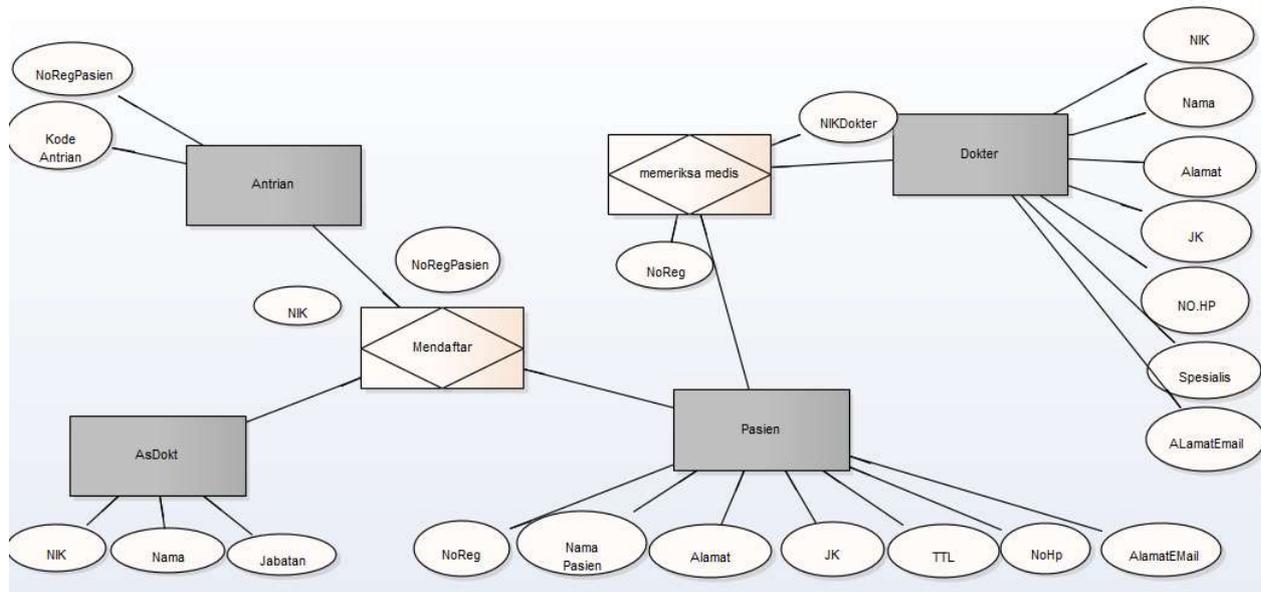
digunakan untuk menyimpan informasi Jadwal Dokter

Tabel 50 Tabel Jadwal Dokter

Nama Field	Tipe
NIK	Integer
Jadwal Wajib	Date & Time
Jadwal Khusus	Date & Time

1.7.1 Entity Relationship Data





Gambar 61 Entity Relationship Diagram

1.7.2 Design Constraints

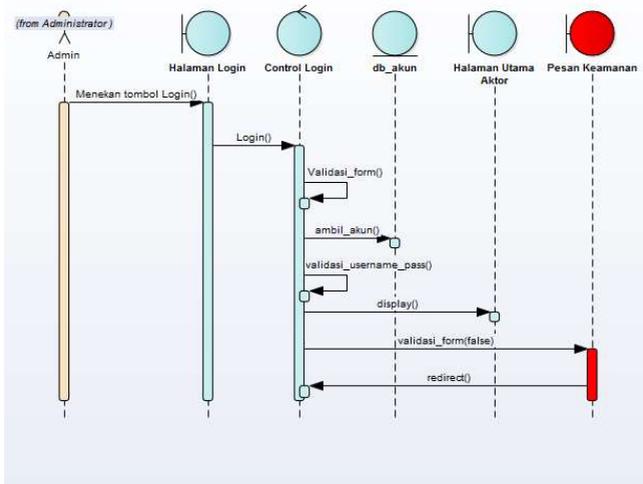
1. System aplikasi android di develop menggunakan Android Studio
2. Komunikasi antara portal aplikasi web dengan database menggunakan SQL
3. Tampilan aplikasi web akan di coding dengan HTML
4. Product akan di coding menggunakan PHP
5. System hanya bisa di run pada Sistem Operasi Windows dan Sistem Operasi Android minimal versi 4.0

1.7.3 Logical Database Requirements

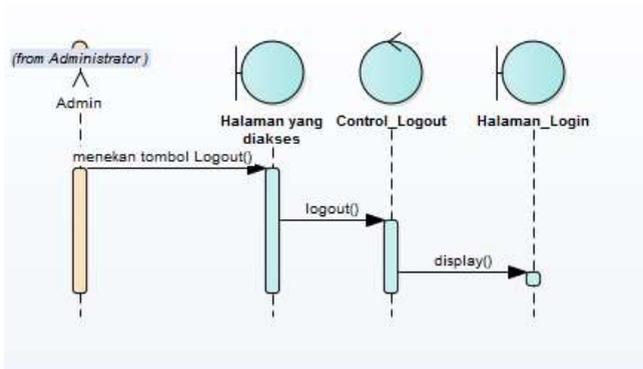
Software system akan dipisah dalam tiga logical database

1. Processor algoritma akan mengurai algoritma ke sebuah struktur tree sebelum melakukan processing. Setelah melakukan processing pada tree, data yang ada dalam dalam sistem tidak akan bisa diakses untuk kedua kalinya.
2. Database akan menyimpan option yang dipilih oleh user. Informasi yang telah disimpan akan dapat diakses oleh hampir semua user interface dengan possible user (user yang telah terotorisasi) dan bisa diakses oleh hampir tiap internal system.
3. Semua data akan disimpan dalam database yaitu username accounts dan profil account, informasi obat, informasi dokter, informasi apotek dll. Database memungkinkan untuk dapat diakses secara bersamaan oleh beberapa user sekaligus dalam satu waktu, membutuhkan design database yang bagus.

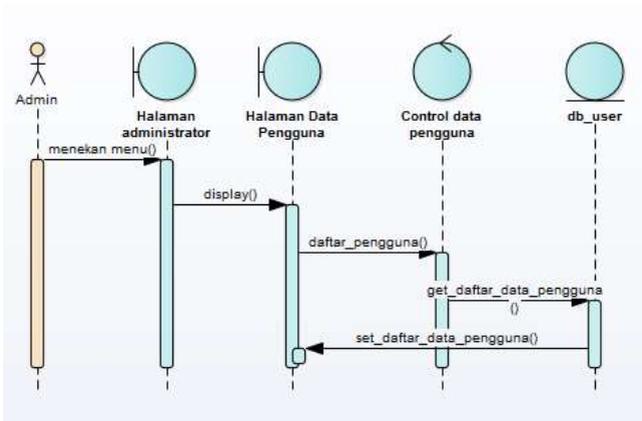
1.7.4 Sequence Diagrams



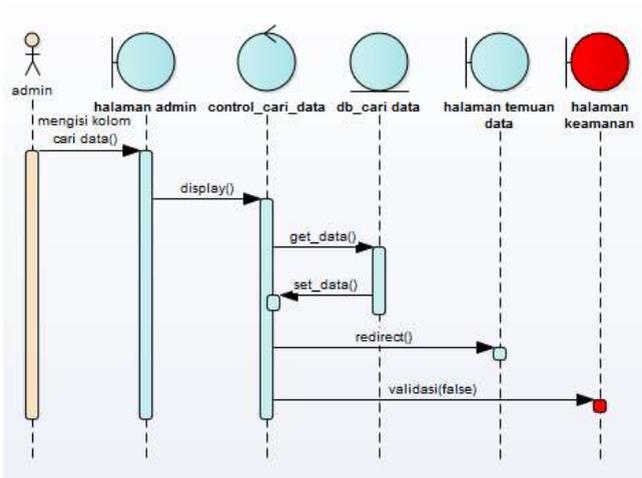
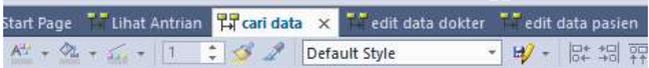
Gambar 62 SD Login



Gambar 63 SD LogOut

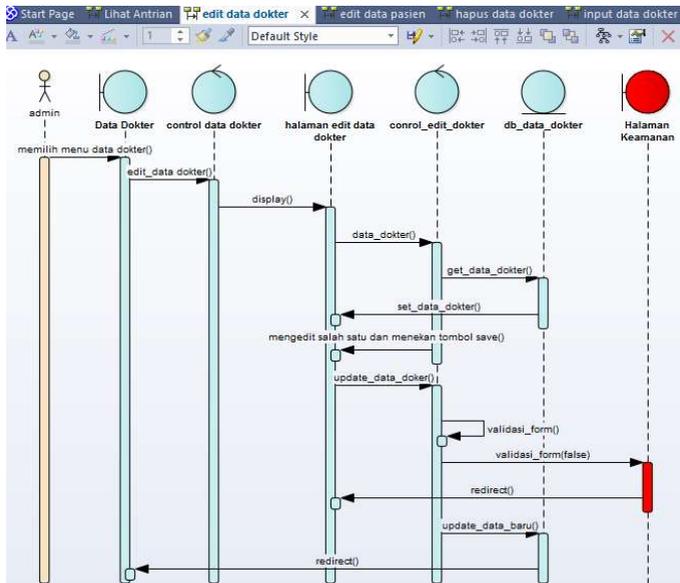


Gambar 64 SD Lihat Daftar Data Pengguna

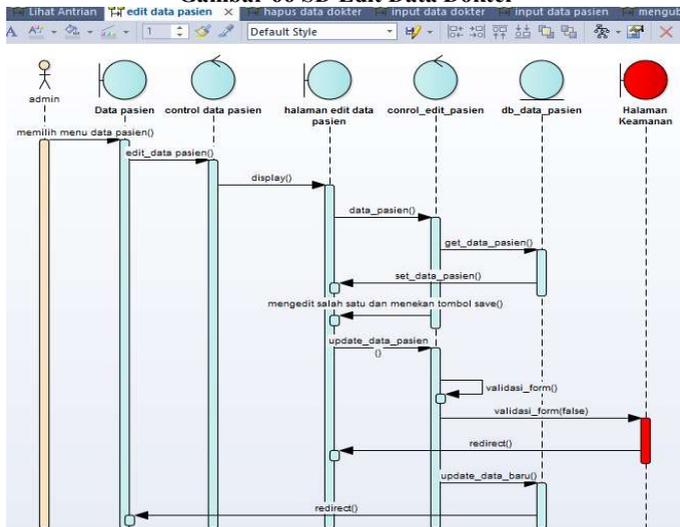


Gambar 65 SD Cari Data

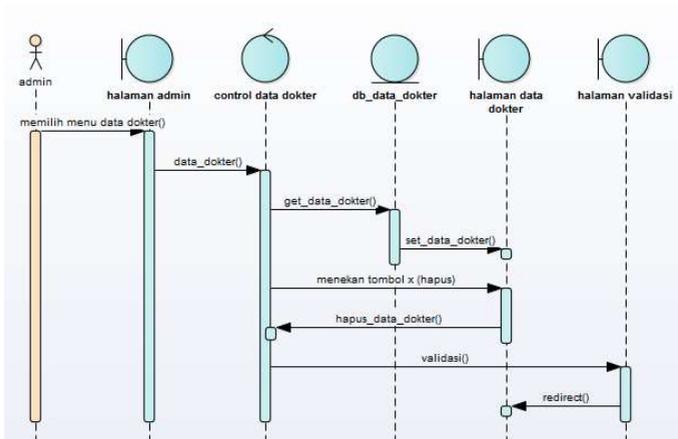
Spesifikasi Kebutuhan Perangkat Lunak



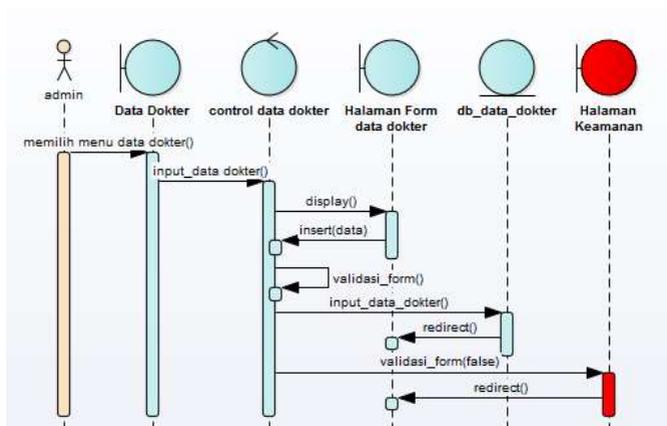
Gambar 66 SD Edit Data Dokter



Gambar 67 SD Edit Data Pasien

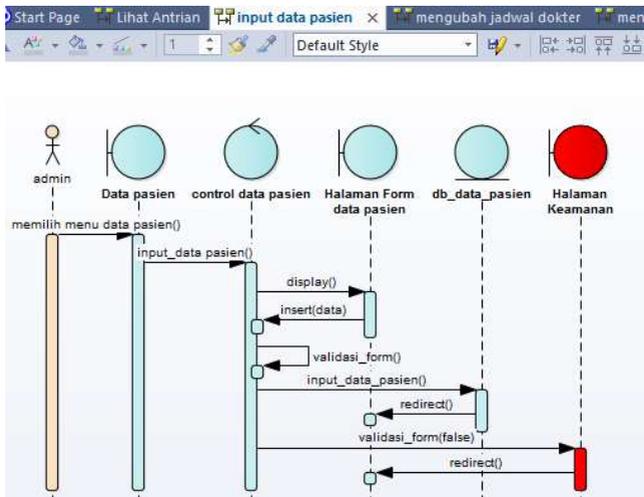


Gambar 68 SD Hapus Data Dokter

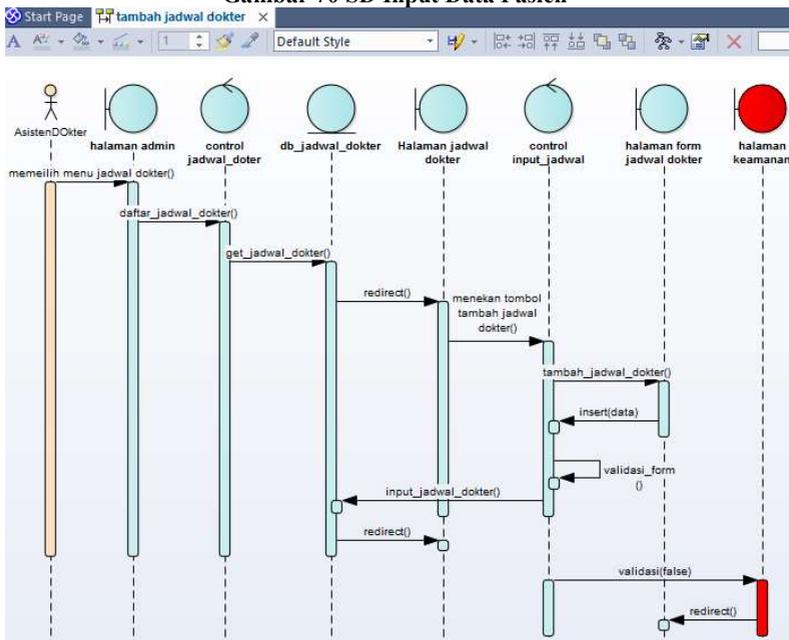


Gambar 69 SD Input Data Dokter

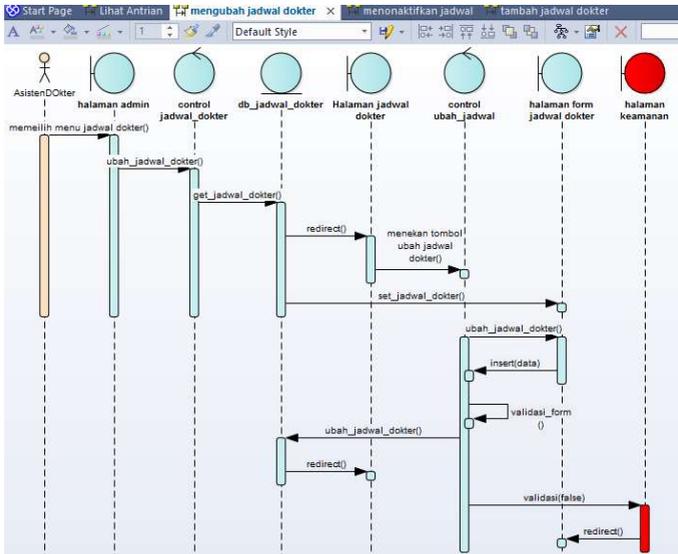
Spesifikasi Kebutuhan Perangkat Lunak



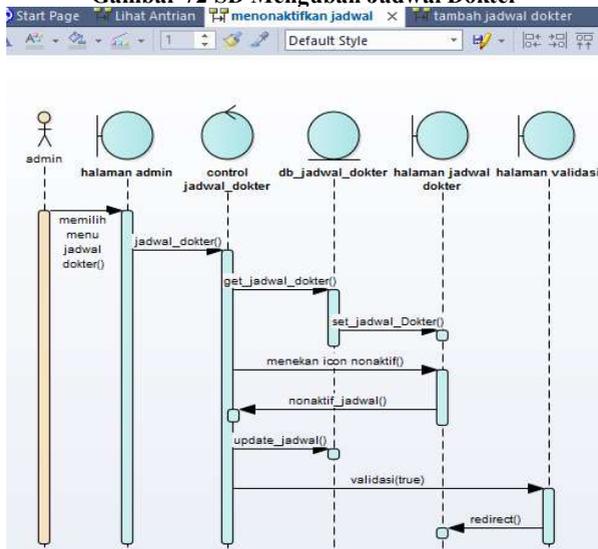
Gambar 70 SD Input Data Pasien



Gambar 71 SD Tambah Jadwal Dokter

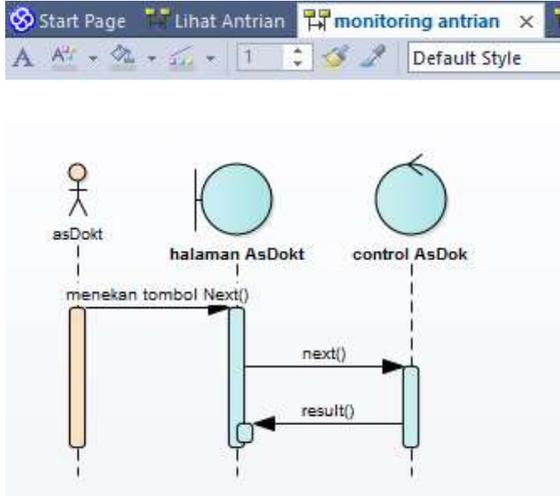


Gambar 72 SD Mengubah Jadwal Dokter

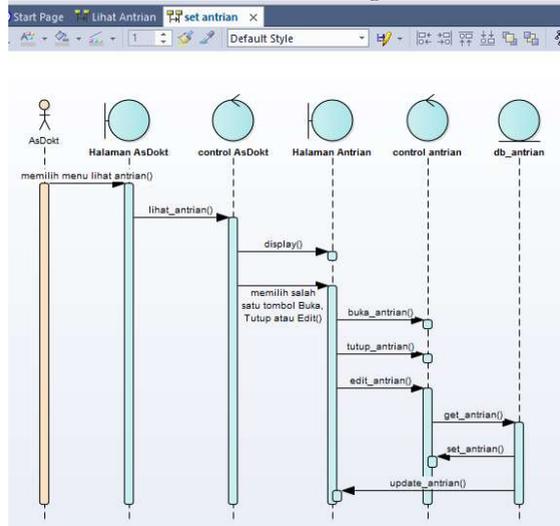


Gambar 73 SD Menonaktifkan Jadwal Dokter

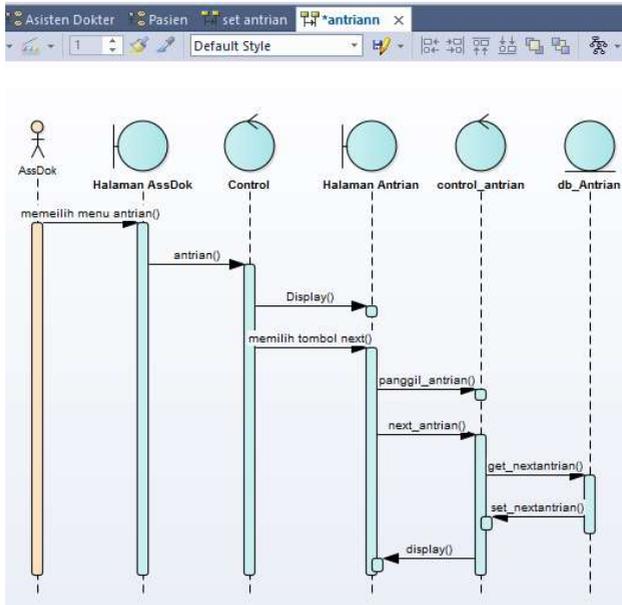
Spesifikasi Kebutuhan Perangkat Lunak



Gambar 74 SD Monitoring Antrian

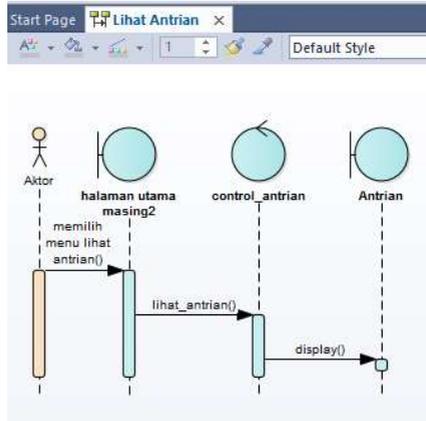


Gambar 75 SD Set Antrian

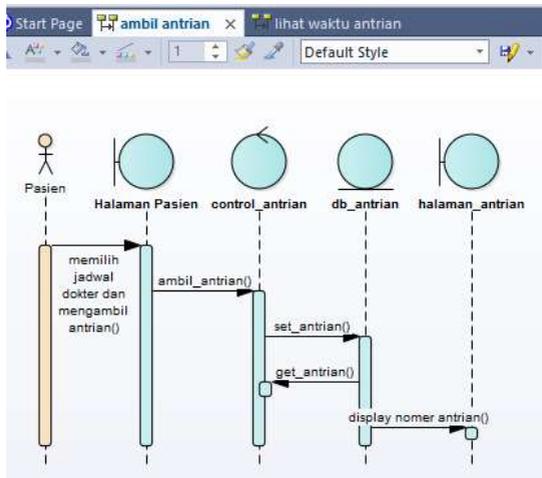


Gambar 76 SD Antrian

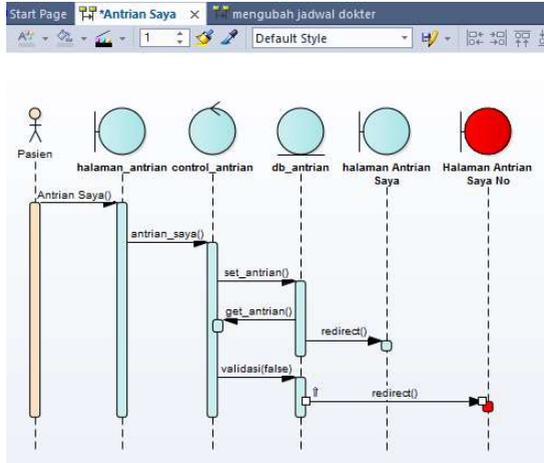
Spesifikasi Kebutuhan Perangkat Lunak



Gambar 77 SD Lihat Antrian



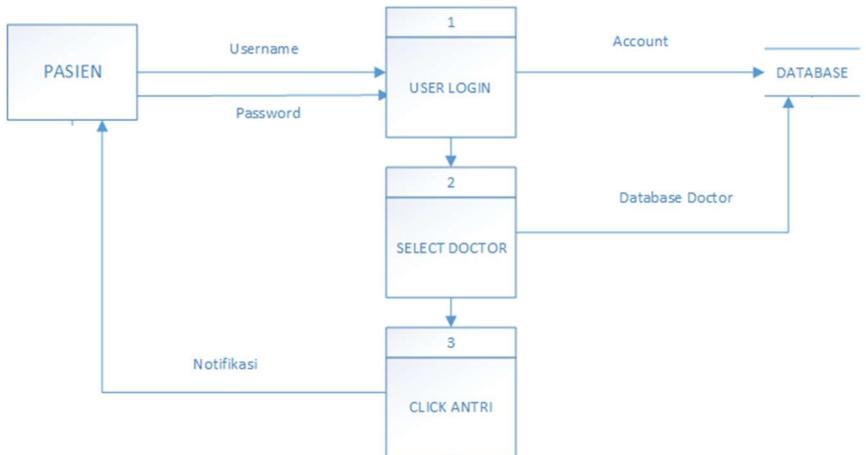
Gambar 78 SD Ambil Antrian



Gambar 79 SD Antrian Saya

1.7.5 Class Diagram

1.7.5.1 Data Flow Diagrams (DFD)



Gambar 80 Data Flow Diagram

1.8 Change Management Process

Untuk memasukkan perubahan dalam requirements dan untuk melihat apa perubahan yang tepat untuk

requirement sistem ini, change management process akan dilakukan secara iterative dengan semua aktivitas dan beberapa strategi tersebut.

Stakeholder akan mengidentifikasi sebuah requirement baru atau memodifikasi sebuah requirement dan mengkomunikasikannya dengan tim leader melalui email atau rapat dengan seluruh anggota team project. Team akan menilai pengerjaan perubahan yang diusulkan dengan memperhitungkan batasan waktu dan batasan struktural dari modul sebelumnya dan mengimplementasikan strategi dan menyerahkan ke mentor untuk mendapatkan persetujuan.

Sebuah rencana perubahan akan dibuat untuk implementasi dari perubahan dan disertai dengan rangkaian perubahan yang telah ditetapkan. Selanjutnya team akan melanjutkan implementasi requirements baru tersebut. Lalu melakukan testing pada program yang telah diperbarui. Selama fase testing, software akan di test untuk memuaskan Request change. Setelah perubahan selesai, selanjutnya akan dilakukan update serta membuat dokumentasi. Perubahan akan diverifikasi untuk terakhir kalinya, dan sebuah software system baru siap untuk dirilis.