

LAPORAN KERJA PRAKTEK
PEMBUATAN APLIKASI ADMISI UNTUK SISTEM INFORMASI
MANAJEMEN RS PKU MUHAMMADIYAH WONOSOBO

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika



Disusun oleh :

Okki Putrawan

09650058

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA
YOGYAKARTA

2012

PENGESAHAN LAPORAN KERJA PERAKTEK

PEMBUATAN APLIKASI ADMISI UNTUK SISTEM INFORMASI

MANAJEMEN RS PKU MUHAMMADIYAH WONOSOBO

Disusun oleh :

Nama : Okki Putrawan

Nim : 09650058

Telah diseminarkan pada tanggal : 4 Januari 2013

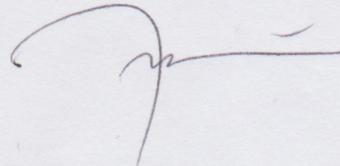
Pembimbing,



Agung Fatwanto, Ph. D.

NIP. 19770103 200501 1 003

Penguji,



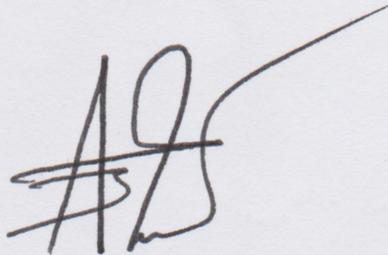
Aulia Faqih Rifa'I, S.Kom, M.Kom.

NIP. 19860306 201101 1 009

Mengetahui,

a.n Dekan

Ketua Program Studi



Agus Mulyanto, S.Si., M.Kom.

NIP. 19710823 199903 1 003

KATA PENGANTAR

Bismillahirrahmanirrahim.

Alhamdulillah, puji syukur kehadirat Allah SWT yang telah melimpahkan segala rahmat dan karunia-Nya, sehingga penulis dapat melakukan Kerja Praktek yang menjadi salah satu syarat untuk menyelesaikan studi di Teknik Informatika UIN Sunan Kalijaga Yogyakarta.

Penulis menyadari bahwa dalam proses pengerjaan Kerja Praktek ini, penulis tidak lepas dari peran berbagai pihak yang telah memberikan dukungan, bantuan dan dorongan sehingga Kerja Praktek ini dapat terlaksana. Dalam kesempatan ini, penulis ingin menyampaikan ucapan terima kasih kepada :

1. Bapak Agus Mulyanto, S.Si.,M.Kom. selaku Kaprodi Teknik Informatika.
2. Bapak Agung Fatwanto, S.Si, M.Kom., Ph.D. selaku dosen pembimbing Kerja Praktek Kelompok 1 yang telah memberikan bimbingan, arahan, motivasi, dan bantuan dalam pengerjaan Kerja Praktek ini.
3. Seluruh dosen Teknik Informatika yang telah memberikan ilmu dan pengalaman kepada penulis.
4. Dra. Hj. Dwi Pudjaningsih, MMR, Apt. selaku pembimbing lapangan yang telah memberikan banyak bantuan, pengarahan dan pengalaman kepada kami.

5. Bapak dan Ibu sebagai orang tua kandung penulis, serta kakak, adik dan Mas yang telah memberikan do'a, dukungan lahir batin dan materi, motivasi, kasih sayang dan cinta kepada penulis.
6. Teman-teman Kelompok 1 Kerja Praktek yang selalu memotivasi, membantu, dan memberikan kerja keras untuk hasil maksimal kita.
7. Teman-teman Teknik Informatika angkatan 2009 yang selalu memberikan motivasi dan semangat kepada kami.
8. Serta pihak-pihak lain yang belum disebutkan penulis yang telah memberikan bantuan sehingga Kerja Praktek ini dapat terlaksana.

Penulis menyadari ketidaksempurnaan dalam pengerjaan Kerja Praktek ini. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat memperbaiki kualitas penulisan ini sehingga dapat bermanfaat.

Yogyakarta, 15 Desember 2012

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan masalah.....	3
1.3 Pembatasan Masalah	3
1.4 Maksud dan Tujuan Kerja Praktek	3
1.5 Manfaat Kerja Peraktek	4
BAB II TEMPAT KERJA PRAKTEK	5
2.1 Gambaran Umum Instansi	5
2.2 Ruang Lingkup Kerja Praktek	8
BAB III HASIL DAN PEMBAHASAN	9
3.1 Analisis	9
3.1.1 Kondisi Kerja Instansi	9
3.1.2 Kondisi Sumber Daya Manusia	10
3.2 Kegiatan Kerja Praktek	10

3.2.1 Analisis Kebutuhan.....	11
3.2.1.1 Analisis Kebutuhab Input.....	11
3.2.1.2 Analisis Kebutuhan Proses.....	11
3.2.1.3 Analisis Kebutuhan Output	12
3.2.1.4 Analisis Kebutuhan Antar Muka	13
3.2.2 Perancangan Perangkat Lunak.....	14
3.2.2.1 Metode Perancangan	14
3.2.2.2 Spesifikasi Kebutuhan.....	14
3.2.2.2.1 Pelayanan Pendaftaran Pasien.....	15
3.2.2.2.2 Mendaftarkan Pasien.....	16
3.2.2.2.3 Mendaftarkan Pasien Baru	18
3.2.2.2.4 Mendaftarkan Pasien Lama.....	19
3.2.2.2.5 Entri Data Demografi Pasien	21
3.2.2.2.6 Entri Data Pendaftaran Pasien.....	22
3.2.2.2.7 Mencari Data Pasien	25
3.2.2.2.8 Mencari Data Pendaftaran Pasien	26
3.2.2.3 Arsitektur Aplikasi	30
3.2.2.3.1 <i>Backend Application</i>	30
3.2.2.3.1.1 Web Service	30
3.2.2.3.1.2 Django	32
3.2.2.3.2 Model Data	34

3.2.2.3.2.1 Model Pendaftaran.....	34
3.2.2.3.2.2 Model Demografi.....	37
3.2.2.3.2.3 Model Unit Layanan	43
3.2.2.3.2.4 Model Konfigurasi.....	44
3.2.2.3.3 Implementasi.....	44
3.2.2.3.3.1 Implementasi Dada Pendaftaran	45
3.2.2.3.3.2 Implementasi Data Demografi	46
3.2.2.3.3.3 Implementasi Data Unit Layanan	47
3.2.2.3.3.4 Implementasi Data Konfigurasi	47
BAB IV PENUTUP	48
4.1 Kesimpulan	48
4.2 Saran.....	48
DAFTAR PUSTAKA	49
LAMPIRAN	

DAFTAR GAMBAR

Gambar 3.1: Alur aktifitas pelayanan pendaftaran pasien	15
Gambar 3.2: Kasus-kasus penggunaan pada aktifitas pendaftaran pasien.....	15
Gambar 3.3 Alur aktifitas mendaftarkan pasien.....	16
Gambar 3.4 Kasus-kasus penggunaan pada aktifitas mendaftarkan pasien.....	17
Gambar 3.5 Alur aktifitas mendaftarkan pasien baru	18
Gambar 3.6 Kasus-kasus penggunaan pada aktifitas mendaftarkan pasien baru	18
Gambar 3.7 Alur aktifitas mendaftarkan pasien lama.....	19
Gambar 3.8 Kasus-kasus penggunaan pada aktifitas mendaftarkan pasien lama	20
Gambar 3.9 Alur aktifitas entry data demografi pasien	22
Gambar 3.10 Alur aktifitas entry data pendaftaran pasien.....	23
Gambar 3.11 Alur aktifitas mencari data pasien	25
Gambar 3.12 Kasus-kasus penggunaan pada aktifitas mencari data pasien	25
Gambar 3.13 Alur aktifitas mengkonfirmasi pendaftaran pasien	27
Gambar 3.14 Kasus-kasus penggunaan pada aktifitas mengkonfirmasi pendaftaran pasien	27

Gambar 3.15 Alur aktifitas edit data demografi pasien	28
Gambar 3.16 Alur aktifitas mencari data pendaftaran pasien	28
Gambar 3.17 Kasus-kasus penggunaan pada mencari data pendaftaran pasien .	29
Gambar 3.18 Implementasi data pendaftaran.....	45
Gambar 3.19 Implementasi data pendaftaran.....	46
Gambar 3.20 Implementasi data Unit Layanan.....	47
Gambar 3.21 Implementasi data konfigurasi	47

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemrosesan informasi adalah salah satu aktivitas yang penting didalam sebuah organisasi. Kecepatan Pemrosesan informasi seringkali terlambat dikarenakan hal - hal praktis yang dikerjakan secara manual. Adanya suatu prosedur kerja berdasarkan struktur organisasi dan konsep manajemen yang diberlakukan terkadang tidak efisien dan dapat berakibat tidak terintegrasinya informasi yang berpengaruh langsung pada kualitas outputnya. Proses manual dalam Pemrosesan informasi terkadang cukup banyak menghabiskan waktu dan biaya(Putra, 2012).

Manajemen sistem informasi adalah teknik pengolahan informasi dalam suatu organisasi. Informasi yang cepat, tepat dan akurat akan mendukung roda perekonomian organisasi tersebut untuk menjadi sehat dan berkembang pesat. Seperti menurut Barry E. Cushing (1974:8) yaitu manajemen sistem informasi berguna mengumpulkan dan mengolah data untuk menghasilkan informasi yang berguna untuk semua tingkatan manajemen di dalam kegiatan perencanaan dan pengendalian.

Dalam perkembangan dilapangan seringkali menimbulkan permasalahan termasuk dalam manajemen sistem informasi rumah sakit khususnya Rumah Sakit PKU Muhammadiyah Wonosobo. Sebagai rumah

sakit yang baru akan berdiri, diperlukan strategi pengembangan manajemen sistem informasi yang tepat sehingga dapat menunjang proses administrasi rumah sakit dan meningkatkan efisiensi sumber daya baik tenaga dan biaya serta waktu. Dalam perkembangannya disiplin ilmu manajemen informasi memperoleh bentuk – bentuk baru yang sebelumnya tidak nampak menjadi terbuka untuk digali dan dikembangkan menjadi teori-teori yang kemudian diaplikasikan untuk menjadi *software* yang dapat berguna untuk mendukung kinerja manusia.

Dalam pelaksanaannya, di dalam sebuah tim pengembangan sistem informasi rumah sakit telah memiliki seorang analisis sistem yang telah merancang dan mengidentifikasi masalah – masalah yang ada. Untuk mengimplementasikan desain software yang telah dirancang oleh analisis sistem, dibutuhkan programmer yang bertugas untuk merealisasikan rancangan tersebut.

Kerja praktek merupakan sarana latihan dalam pengembangan pendidikan yang diperoleh dibangku kuliah sehingga dengan adanya kerja praktek ini diharapkan mahasiswa memahami bagaimana menerapkan ilmunya di lapangan pekerjaan. Dan melihat dari penjabaran di atas, pada kerja praktek kali ini kami melibatkan dalam pengembangan sistem informasi rumah sakit sebagai programmer atau pengembang.

1.2 Perumusan Masalah

Pada setiap Organisasi dalam bentuk apapun untuk mewujudkan tujuannya tentu tidak lepas dari berbagai masalah yang harus dihadapi oleh Organisasi tersebut.

1. Kecepatan Pemrosesan informasi dalam administarsi pendaftaran pasien sering kali terlambat dikarenakan hal - hal yang dikerjakan secara manual.
2. Sistem Penyimpanan data yang masih manual sangat menyulitkan dan tidak efisien,khususnya dalam penyimpanan dan pengaksesan data pasien.

1.3 Pembatasan Masalah

Dalam permasalahan yang ada mengenai manajemen sistem informasi pada Rumah Sakit PKU Muhammadiyah Wonosobo, kami mengfokuskan pada aplikasi admisi didalam sistem informasi manajemen rumah sakit tersebut.

1.4 Maksud dan Tujuan Kerja Praktek

Dengan adanya kerja praktek ini ditujukan agar mahasiswa dapat :

1. Meningkatkan kinerja rumah sakit dalam proses pendaftaran pasien.
2. Mempermudah kariawan rumah sakit untuk mengakses data pasien.

1.5 Manfaat Kerja Praktek

Diharapkan dari pelaksanaan kerja praktek ini dapat membawa manfaat bagi beberapa pihak baik dari Rumah Sakit PKU Muhammadiyah Wonosobo maupun bagi mahasiswa sendiri.

1.5.1 Manfaat bagi Rumah Sakit PKU Muhammadiyah Wonosobo

1. Peningkatan kinerja rumah sakit dalam proses pendaftaran pasien dapat meningkatkan pelayanan pendaftaran pasien menjadi lebih baik.
2. Pengaksesan data pasien yang lebih mudah dapat meningkatkan kinerja karyawan dalam melayani pendaftaran pasien.

1.5.2 Manfaat bagi Mahasiswa

1. Mahasiswa memperoleh pengalaman kerja sebelum memasuki dunia kerja.
2. Mahasiswa dapat melatih dirinya untuk dapat berorganisasi dan bersosialisasi dengan lingkungan kerja.
3. Mahasiswa memperoleh kemampuan untuk mengembangkan aplikasi sistem informasi.

BAB II

TEMPAT KERJA PRAKTEK

2.1 Gambaran Umum Instansi

CV. Medcominfo Mitra Solusi merupakan perusahaan yang bergerak di bidang teknologi informasi, pengembangan sumber daya manusia, konsultan bidang kesehatan, dan penyaluran tenaga ahli di bidang kesehatan.

Perusahaan yang didirikan pada tanggal 24 September 2006 ini dirancang atas dasar filosofi bahwa pada dasarnya orang diciptakan hanya untuk mengabdikan kepada Tuhan, orang diciptakan untuk beribadah kepada Tuhan, dan orang yang baik adalah orang yang berguna bagi orang lain. Didasarkan pula pada pemikiran bahwa kedepan teknologi informasi merupakan pilihan utama dan menjadi prioritas penyelesaian masalah. Manusia bersumber daya merupakan aset yang menentukan perlu diperhatikan lebih serius, jasa konsultan, dan perlu penyaluran tenaga kerja kesehatan. Bergai landasan pikir di ataslah yang menjadi tolak ukur didirikannya perusahaan ini.

Prioritas utama perusahaan kami adalah fokus pada pelayanan bidang kesehatan dan semua hal yang terkait utama dalam bidang farmasi. Harapan kami produk yang kami hasilkan akan menjadikan solusi

berkualitas bagi masalah anda dalam bidang kesehatan dan bermanfaat bagi umat manusia.

Visi dari CV. Medcominfo Mitra

Terwujudnya perusahaan terkemuka berbasis pengetahuan dan teknologi dengan layanan terbaik dalam memberikan solusi terpercaya dan bernilai tambah bagi pelanggan dan stakeholder.

Misi dari CV. Medcominfo Mitra

- Menghasilkan produk dan jasa berbasis pengetahuan dan teknologi yang mampu bersaing di pasar dan bermanfaat bagi masyarakat
- Menghasilkan produk dan jasa dalam bidang farmasi dan kesehatan dengan orientasi bisnis dan kultur profesional
- Meningkatkan benefit dan value bagi pelanggan dan stekholder
- Menutamakan layanan terbaik dan solusi terpercaya bagi kepuasan pelanggan dan stakeholder

Tata Nilai dan Budaya Kerja Perusahaan

- Kami menomor satukan etika bisnis.
- Kami selalu menghasilkan produk berkualitas.

- Kami menghargai dan menjunjung tinggi Integritas dan profesionalitas kerja
- Kami berkomitmen kepada layanan terbaik dan kepuasan pelanggan

Tujuan

- Menghasilkan produk dan jasa berbasis pengetahuan dan teknologi informasi yang mampu bersaing di pasar dan bermanfaat bagi masyarakat
- Menghasilkan produk dan jasa dalam bidang farmasi dan kesehatan dengan orientasi bisnis dan kultur profesional
- Meningkatkan benefit dan value bagi pelanggan dan stakeholder
- mengutamakan layanan terbaik dan solusi terpercaya bagi kepuasan pelanggan dan stakeholder

Strategi Perusahaan

- Memelihara keseimbangan antara pertumbuhan dan profitabilitas
- Melaksanakan operational excellence
- Mengembangkan intellectual capital
- Mencetak sumber daya manusia yang berkualitas dan dapat diterima oleh pasar tenaga kerja
- Mengembangkan dan menghasilkan industri teknologi informasi yang memberikan nilai tambah, mendukung layanan optimal, dan

meningkatkan profitabilitas perusahaan pelanggan

- Menjalinkan kerjasama, konsultasi manajerial, dan aliansi strategi yang saling menguntungkan
- Membudayakan inovasi dan belajar tiada henti (*long life learning*) yang dapat meningkatkan produktivitas secara nyata (*significant*).

2.2 Ruang Lingkup Kerja Praktek

CV. Medcominfo Mitra adalah perusahaan yang bergerak di bidang teknologi informasi khususnya dalam pengembangan aplikasi khusus kesehatan. Pada kegiatan kerja praktek kali ini, kami terlibat dalam proses pengembangan SIMRS PKU MUHAMMADIAH WONOSOBO yang merupakan proyek dari perusahaan ini.

BAB III

HASIL DAN PEMBAHASAN

3.1 Analisis

3.1.1 Kondisi Kerja Instansi

Instansi ini memiliki sebuah kantor sebagai tempat koordinasi dalam pengerjaan proyek SIMRS PKU Muhammadiyah Wonosobo. Untuk memudahkan tim pengembang dalam melakukan pekerjaannya dan koordinasi pekerjaan, perusahaan memfasilitasi tim pengembang dengan membuat sebuah akun **Github** untuk meletakkan file-file yang berhubungan dengan proyek SIMRS. Anggota tim pengembang dapat melihat dengan mudah file - file yang telah diubah atau file baru yang ditambahkan dalam direktori proyek.

3.1.2 Kondisi Sumber Daya Manusia

Proyek SIMRS PKU Muhammadiyah Wonosobo yang dikembangkan oleh **CV. Medcominfo** ini ditangani oleh tim pengembang yang dibentuk oleh perusahaan khusus untuk mengembangkan sistem ini. Tim ini terdiri dari satu Sistem Analis dan 3 Programmer. Berikut anggota tim pengembang SIMRS PKU Muhammadiyah Wonosobo.

Direktur : Dra. Sri Pudjaningsih,

Sistem Analis : Firmansyah Adiputra, S.Kom

Programmer : 1. Arvin Nizar, S.Kom
2. Uji Setyawan, S.Kom
3. Joko Munandar

3.2 Kegiatan Kerja Praktek

Dalam pengembangan SIMRS, ada beberapa aplikasi yang dibuat diantaranya aplikasi admisi, aplikasi inventori, aplikasi farmasi, dan lain - lain. Dalam kerja praktek ini, kami dipercaya untuk mengerjakan aplikasi admisi, yaitu aplikasi untuk manajemen pendaftaran pasien. Proses bisnis dan analisis aplikasi admisi telah dibuat oleh sistem analis, sehingga kami tinggal mengimplementasikannya. Berdasarkan user requirement, sistem requirement dan analisa sistem yang telah disusun oleh analis, aplikasi admisi mempunyai dua modul. Modul – modul yang ada dalam aplikasi admisi yaitu :

- Modul Demografi : Modul yang berfungsi untuk menyimpan data demografi pasien
- Modul Pendaftaran : Modul ini berfungsi untuk mengelola data pendaftaran pasien.

3.2.1 Analisis Kebutuhan

3.2.1.1 Analisis Kebutuhan Input

Input atau masukan dari aplikasi admisi ini terdiri dari 1 karakteristik masukan yaitu petugas biasa. Input yang dapat dilakukan oleh petugas pendaftaran adalah sebagai berikut:

- Input data demografi
- Input data pendaftaran
- Input *keyword* untuk pencarian data demografi
- Input *keyword* untuk pencarian data pendaftaran

3.2.1.2 Analisis Kebutuhan Proses

Berdasarkan analisis yang telah dilakukan maka dapat diketahui kebutuhan yang akan digunakan dalam pengolahan data dari input data yang diberikan kepada sistem sehingga menghasilkan output yang sesuai dengan yang diharapkan. Kebutuhan proses dalam aplikasi admisi adalah sebagai berikut:

1. Login
2. Manajemen data demografi pasien
3. Manajemen data pendaftaran pasien
4. Pencarian data demografi

5. Pencarian data pasien

3.2.1.3 Analisis Kebutuhan Output

Keluaran yang diperoleh dari aplikasi admisi ini adalah sebagai berikut:

1. Informasi data demografi pasien
2. Informasi data pendaftaran pasien

3.2.1.4 Analisis Kebutuhan Antarmuka

Antarmuka atau *interface* merupakan sarana yang menjadi penghubung antara pengunjung dengan sistem informasi. Kebutuhan antar muka yang dibutuhkan adalah yang memudahkan *user* dalam melakukan akses terhadap sistem, nyaman serta meminimalkan terjadinya kesalahan. Berikut antarmuka yang ada dalam aplikasi admisi ini:

3.2.1.4.1 Antarmuka Sistem

Untuk komponen-komponen lain yang ada dalam SIMRS, Aplikasi Pendaftaran Pasien ini nantinya akan menyediakan data pendaftaran pasien dan hasil skrining awal.

3.2.1.4.2 Antarmuka Pengguna

Antarmuka pengguna yang harus disediakan oleh aplikasi ini adalah sebagai berikut:

1. Entri data demografi pasien
2. Pencarian data demografi untuk pendaftaran pasien
3. Entri data pendaftaran
4. Pencarian data pendaftaran untuk konfirmasi pendaftaran pasien
5. Mencetak nota pendaftaran
6. Mencetak nomor antrian
7. Mencetak kartu registrasi dengan isi sebagai berikut:
no RM, nama pasien, alamat, tahun kunjungan pertama kali.

3.2.2 Perancangan Perangkat Lunak

3.2.2.1 Metode Perancangan

Metode perancangan yang dikembangkan untuk membangun aplikasi ini adalah dengan menggunakan metode perancangan berorientasi objek atau menggunakan UML.

Hasil perancangan aplikasi ini akan dijelaskan secara detail dengan *activity diagram* dan *use case* berdasarkan kebutuhan fungsional yang ada dalam aplikasi ini. Penjelasan lebih detail *use case diagram* dan alur proses aplikasi akan dijelaskan menggunakan *sequence diagram*.

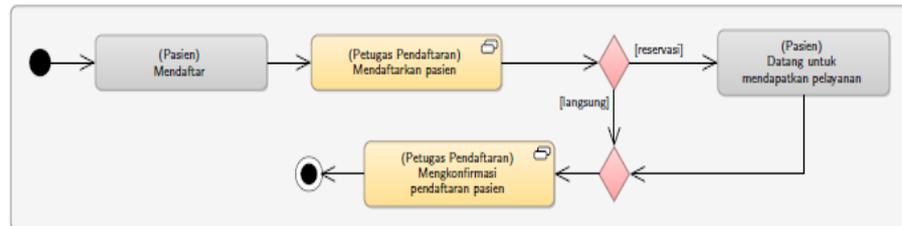
3.2.2.2 Spesifikasi Kebutuhan

Secara umum, fungsi – fungsi yang harus disediakan oleh aplikasi admisi adalah sebagai berikut:

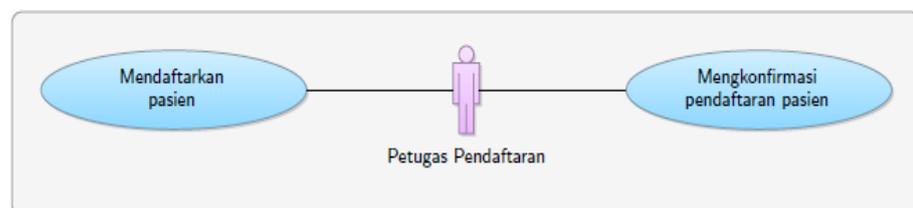
1. Membantu melayani proses pendaftaran pasien
2. Menyediakan data pendaftaran pasien

Berdasarkan fungsi produk yang telah disebutkan di atas, maka kebutuhan fungsional yang harus dipenuhi oleh aplikasi ini akan dijabarkan pada subbab – subbab berikut :

3.2.2.2.1 Pelayanan Pendaftaran Pasien



Gambar 3.1: Alur aktifitas pelayanan pendaftaran pasien



Gambar 3.2: Kasus-kasus penggunaan pada aktifitas pendaftaran pasien

Aplikasi ini nantinya harus melayani kegiatan pelayanan pendaftaran pasien. Pendaftaran pasien dapat dilakukan dengan dua cara, yaitu :

1. Pendaftaran secara langsung, dimana pasien mendaftar secara langsung ke rumah sakit untuk mendapatkan pelayanan hari itu juga.
2. Pendaftaran reservasi, dimana pasien mendaftar melalui via telephon tanpa datang langsung ke rumah sakit atau langsung ke rumah sakit tetapi tidak untuk mendapatkan pelayanan pada hari itu.

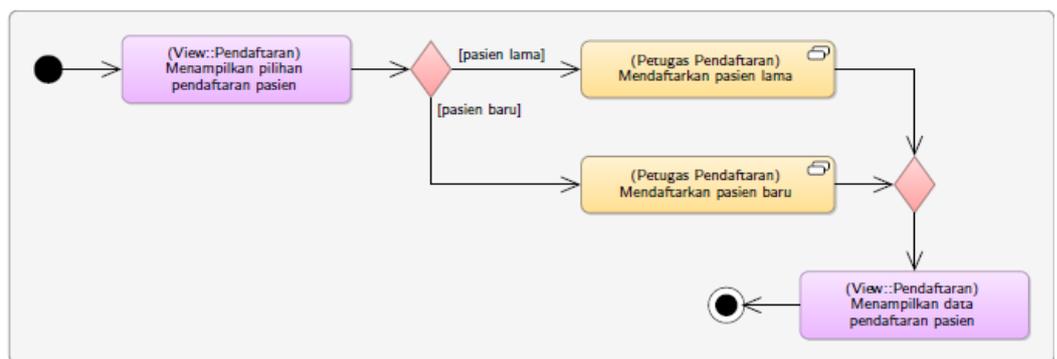
Setiap pendaftaran akan menghasilkan nomor antrian. Saat pasien datang untuk mendapatkan pelayanan pada hari yang telah ditentukan, maka

pendaftaran akan dikonfirmasi untuk dikenai biaya yang dibuktikan dengan dicetaknya nota pendaftaran.

Alur dari kegiatan pelayanan pendaftaran pasien dapat dilihat pada Gambar 3.1, dengan aktifitas yang berwarna coklat adalah aktifitas yang dilakukan dengan bantuan aplikasi ini.

Berdasarkan alur kegiatan pelayanan pendaftaran pasien tersebut, maka dapat dijabarkan kasus-kasus penggunaan utama yang ada dalam aktifitas ini seperti yang digambarkan pada Gambar 3.2. Penjelasan untuk setiap kasus penggunaan ada pada anak subbab berikut.

3.2.2.2.2 Mendaftarkan Pasien



Gambar 3.3: Alur aktifitas mendaftarkan pasien

Pengguna Petugas Pendaftaran.

Tujuan Melayani pendaftaran pasien.

Pemicu Saat pasien melakukan pendaftaran.

Hasil Nomor antrian pelayanan dan tersimpannya data pendaftaran serta data demografi pasien (untuk pasien baru).



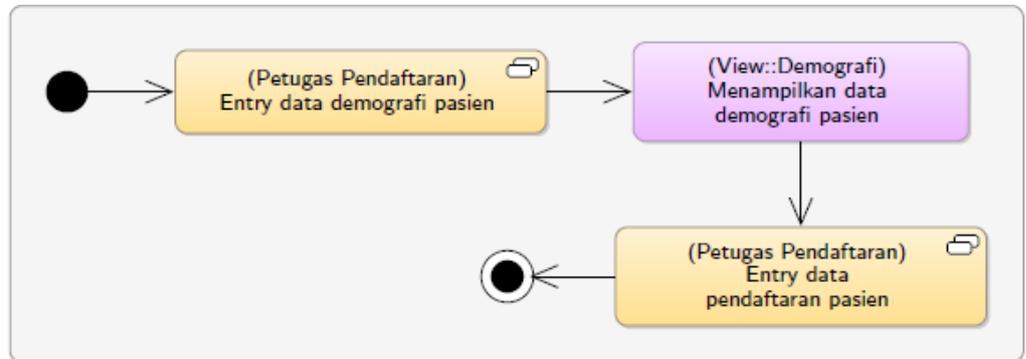
Gambar 3.4: Kasus-kasus penggunaan pada aktifitas mendaftarkan pasien

Deskripsi Aktifitas pendaftaran pasien dibedakan untuk pasien lama dan pasien baru. Untuk pasien baru, perlu dicatat data demografi pasien dan pasien akan mendapatkan kartu registrasi. Alur aktifitas dalam kasus penggunaan ini dapat dilihat pada Gambar 3.3. Berdasarkan alur aktifitas tersebut, kasus penggunaan ini dipecah menjadi beberapa kasus penggunaan lagi seperti yang digambarkan pada Gambar 3.4.

Ekstensi

- Mendaftarkan pasien lama
- Mendaftarkan pasien baru

3.2.2.2.3 Mendaftarkan Pasien Baru



Gambar 3.5: Alur aktifitas mendaftar pasien baru

Pengguna Petugas Pendaftaran.

Tujuan Melayani pendaftaran pasien baru.

Pemicu Saat pasien baru melakukan pendaftaran.

Hasil Nomor antrian pelayanan dan tersimpannya data demografi serta data pendaftaran pasien.



Gambar 3.6: Kasus-kasus penggunaan pada aktifitas mendaftar pasien baru

Deskripsi Untuk pasien baru, sebelum didaftarkan untuk mendapatkan pelayanan,

terlebih dahulu data demografi dari pasien tersebut dicatat dan disimpan.

Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.5. Berdasarkan

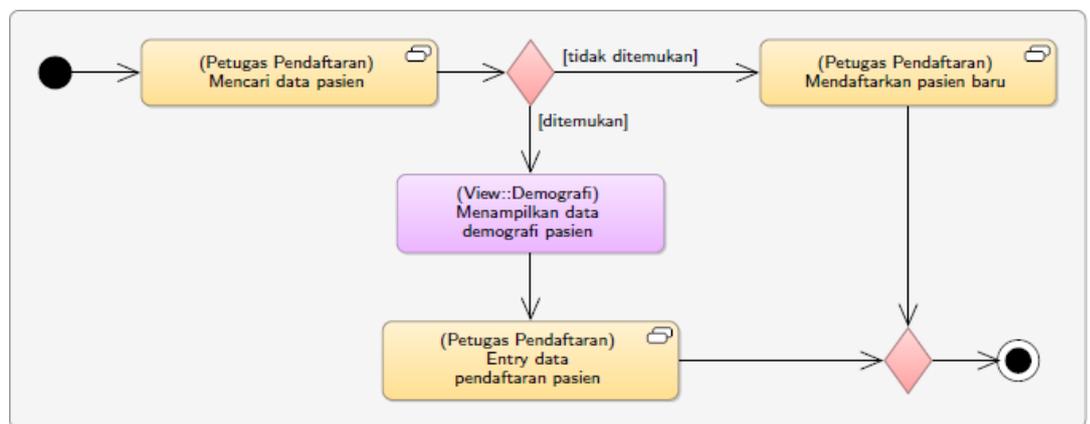
alur aktifitas tersebut, kasus penggunaan ini dipecah menjadi beberapa

kasus penggunaan lagi seperti yang digambarkan pada Gambar 3.6.

Includes

- Entry data demografi pasien
- Entry data pendaftaran pasien

3.2.2.2.4 Mendaftarkan pasien Lama

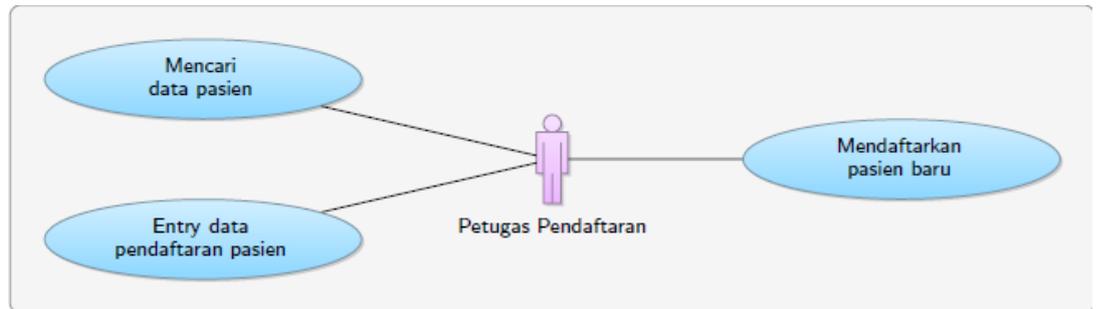


Gambar 3.7: Alur aktifitas mendaftar pasien lama

Pengguna Petugas Pendaftaran.

Tujuan Melayani pendaftaran pasien lama.

Pemicu Saat pasien lama melakukan pendaftaran.



Gambar 3.8: Kasus-kasus penggunaan pada aktifitas mendaftarkan pasien lama

Hasil Nomor antrian pelayanan dan tersimpannya data pendaftaran.

Deskripsi Dalam aktifitas mendaftarkan pasien lama, terlebih dahulu dicari data demografi untuk pasien yang bersangkutan. Bila ditemukan, maka dicatat data pendaftaran untuk pasien tersebut. Namun bila tidak ditemukan, maka pasien akan dianggap sebagai pasien baru. Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.7. Berdasarkan alur aktifitas tersebut, kasus penggunaan ini dipecah menjadi beberapa kasus penggunaan lagi seperti yang digambarkan pada Gambar 3.8.

Includes

- Mencari data pasien
- Entry data pendaftaran pasien (bila data pasien ditemukan)

- **Mendaftarkan pasien baru** (bila data pasien tidak ditemukan)

3.2.2.2.5 Entri Data Demografi Pasien

Pengguna Petugas Pendaftaran.

Tujuan Menyimpan data demografi pasien.

Pemicu Saat pasien baru melakukan pendaftaran.

Hasil Tersimpannya data demografi pasien.

Deskripsi Dalam aktifitas ini, data demografi pasien disimpan dalam dua kategori yaitu utama dan pelengkap. Data utama adalah data yang harus diisikan. Data pelengkap boleh tidak diisikan saat pendaftaran pasien baru, namun saat konfirmasi pendaftaran, data ini harus dilengkapi. Penjelasan tentang data-data yang termasuk dalam kategori data utama dan data pelengkap akan dijelaskan pada [Bab IV](#).

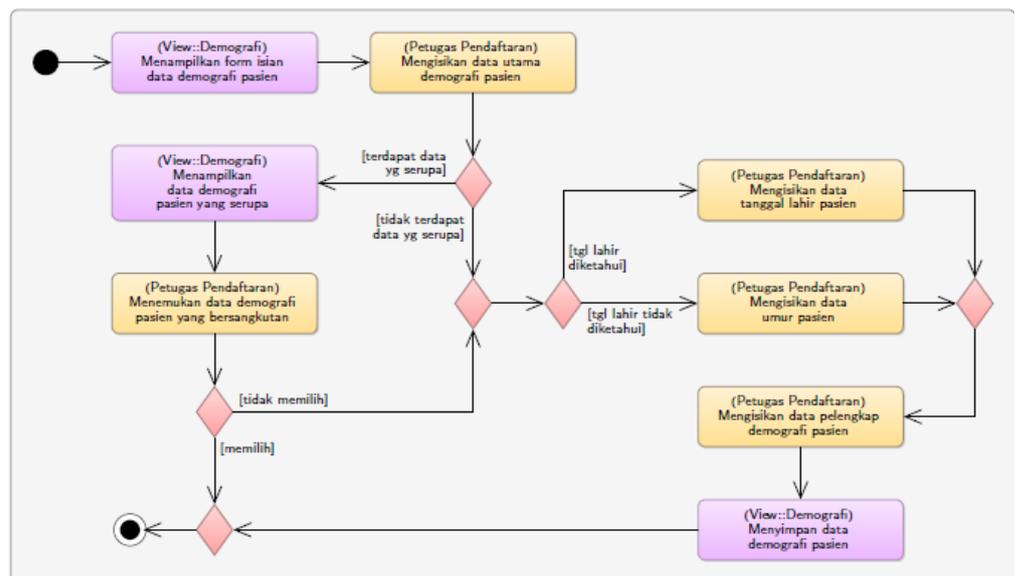
Bila tanggal lahir pasien tidak diketahui, dapat diisikan umur dari pasien dan sistem akan menggunakan tanggal hari ini untuk menebak tanggal lahir pasien.

Saat petugas selesai mengisikan data utama, sistem akan mencari data demografi yang serupa dengan data yang telah diisikan, dan menampilkannya. Untuk menghindari

duplikasi data demografi pasien, petugas sangat dianjurkan untuk

meneliti data demografi yang ditampilkan, dan memilih salah satu dari data tersebut bila memang data tersebut adalah data demografi milik dari pasien yang bersangkutan.

Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.9.



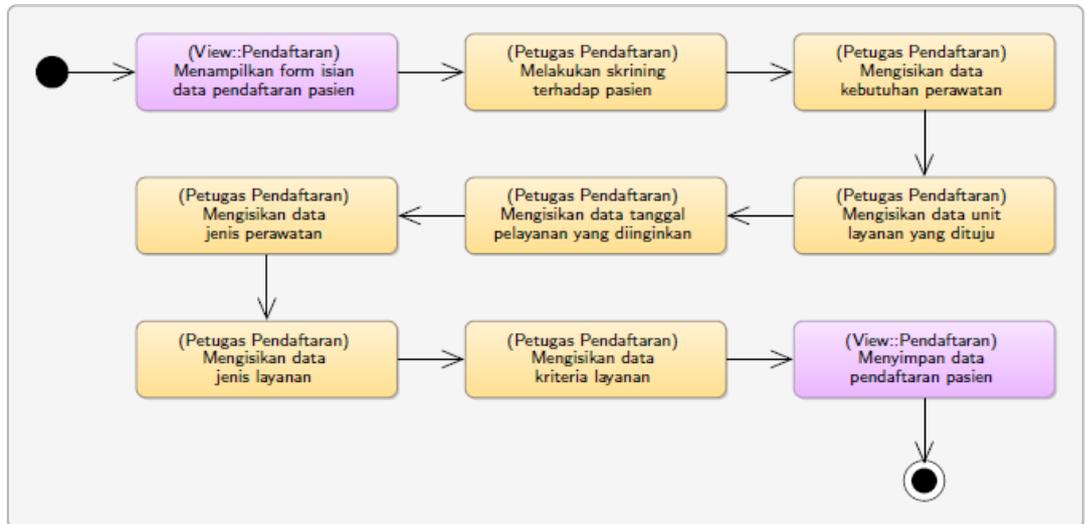
Gambar 3.9: Alur aktifitas entry data demografi pasien

3.2.2.2.6 Entri Data Pendaftaran pasien

Pengguna Petugas Pendaftaran.

Tujuan Menyimpan data pendaftaran pasien.

Pemicu Saat pasien melakukan pendaftaran.



Gambar 3.10: Alur aktifitas entry data pendaftaran pasien

Hasil Nomor antrian pelayanan dan tersimpannya data pendaftaran pasien.

Deskripsi Dalam aktifitas entry data pendaftaran pasien ini, petugas pendaftaran mengisikan data-data sebagai berikut:

- Kebutuhan perawatan (darurat, mendesak, segera, biasa);
- Unit layanan yang dituju dan tanggal pelayanan;
- Jenis perawatan (rawat inap, rawat jalan);
- Jenis layanan (preventif, kuratif, paliatif, rehabilitatif);
- Kriteria layanan (biasa, intensif, khusus).

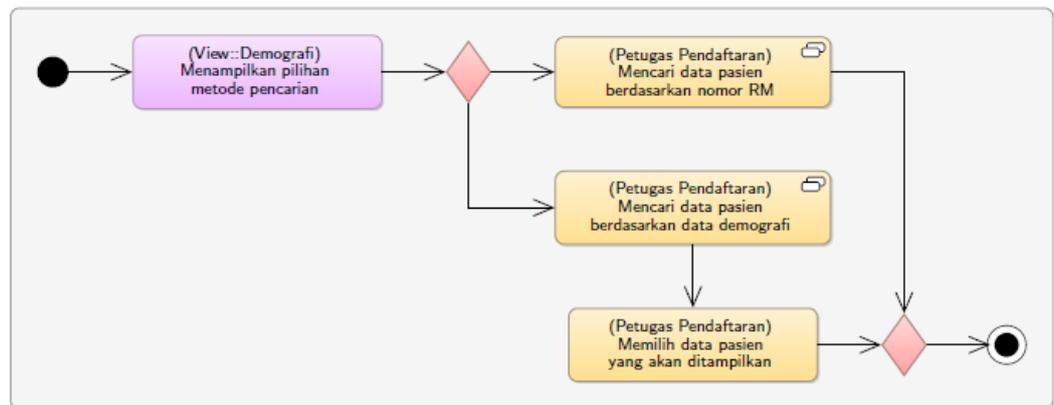
Data tersebut diisikan berdasarkan hasil skrining yang dilakukan oleh petugas pendaftaran. Menurut standar APKP.1 (Joint Commission International, 2010), proses skrining dilakukan untuk mencocokkan kebutuhan pasien dengan misi dan sumber daya rumah sakit. Proses skrining ini dapat melalui kriteria triase, evaluasi visual, pemeriksaan fisik, atau hasil-hasil pemeriksaan sebelumnya yang terkait fisik, psikologi, dan laboratorium klinis, atau evaluasi pencitraan diagnostik. Yang terpenting adalah, keputusan untuk mengobati, memindahkan, atau merujuk dilakukan hanya setelah hasil evaluasi skrining tersedia.

Menurut standar APKP.1.1.2 (Joint Commission International, 2010), ketika pasien dipertimbangkan untuk masuk sebagai pasien rawat inap, penilaian skrining membantu staf mengidentifikasi dan memprioritaskan layanan preventif, kuratif, paliatif, dan rehabilitatif yang dibutuhkan pasien; kemudian ia memilih layanan atau unit yang paling sesuai untuk kebutuhan pasien yang paling mendesak (kebutuhan prioritas).

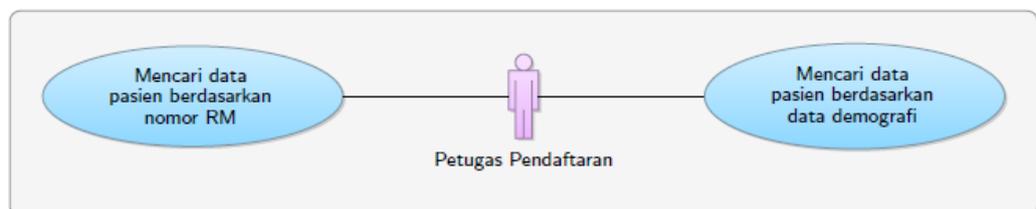
Saat data pendaftaran disimpan, secara otomatis, sistem akan mencatat petugas pendaftaran yang

mengisikan data pendaftaran ini dan menyimpan tanggal hari ini sebagai tanggal pendaftaran. Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.10.

3.2.2.2.7 Mencari Data Pasien



Gambar 3.11: Alur aktifitas mencari data pasien



Gambar 3.12: Kasus-kasus penggunaan pada aktifitas mencari data pasien

Pengguna Petugas Pendaftaran.

Tujuan Menemukan data pasien.

Pemicu Saat data pasien dibutuhkan untuk pendaftaran atau hal lainnya.

Hasil Data pasien hasil pencarian yang dipilih untuk ditampilkan.

Deskripsi Pencarian data pasien dapat dilakukan berdasarkan nomor rekam medik atau berdasarkan data demografi pasien.

Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.11. Berdasarkan alur aktifitas tersebut, kasus penggunaan ini dipecah menjadi beberapa kasus penggunaan lagi seperti yang digambarkan pada Gambar 3.12.

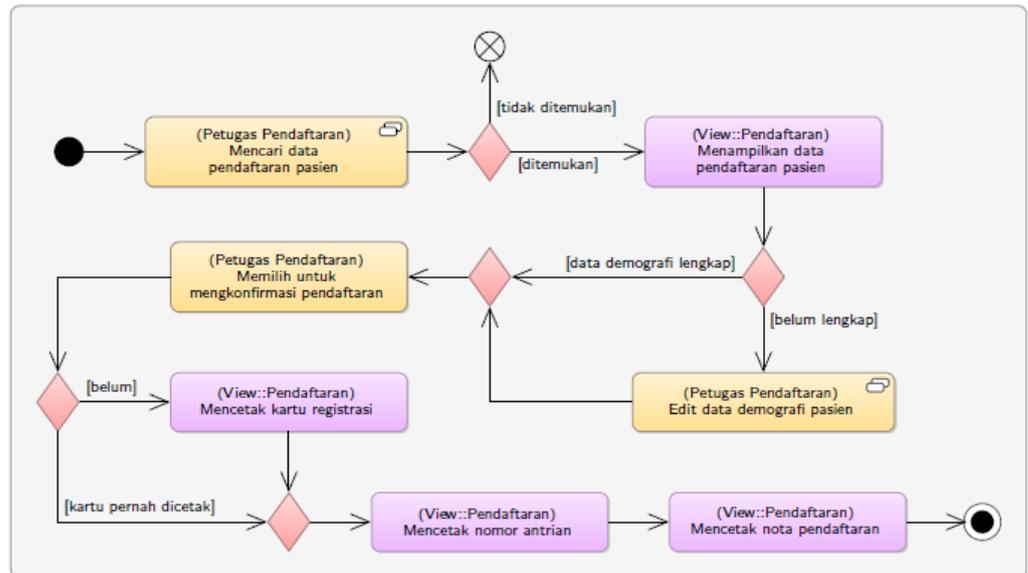
Ekstensi

- Mencari data pasien berdasarkan nomor RM
- Mencari data pasien berdasarkan data demografi

3.2.2.2.8 Mencari Data Pendaftaran Pasien

Pengguna Petugas Pendaftaran.

Tujuan Menemukan data pendaftaran pasien.



Gambar 3.13: Alur aktifitas mengkonfirmasi pendaftaran pasien



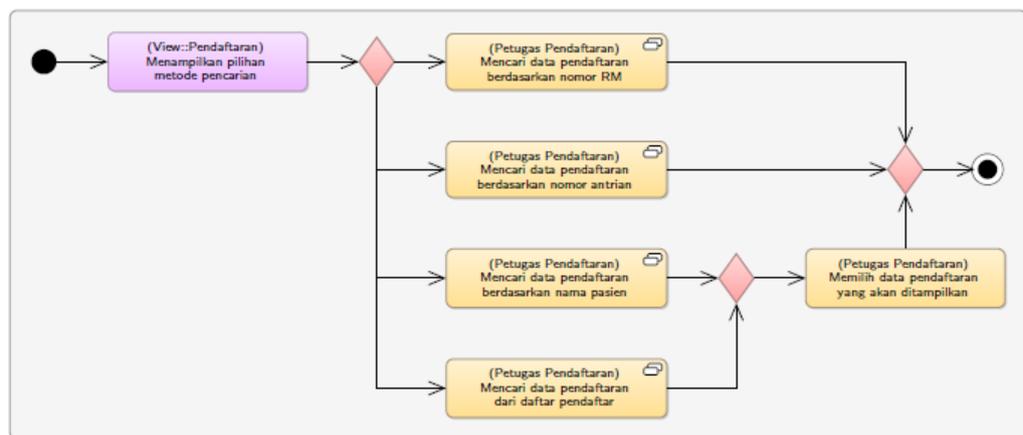
Gambar 3.14: Kasus-kasus penggunaan pada aktifitas mengkonfirmasi pendaftaran pasien

Pemicu Saat akan mengkonfirmasi pendaftaran pasien.

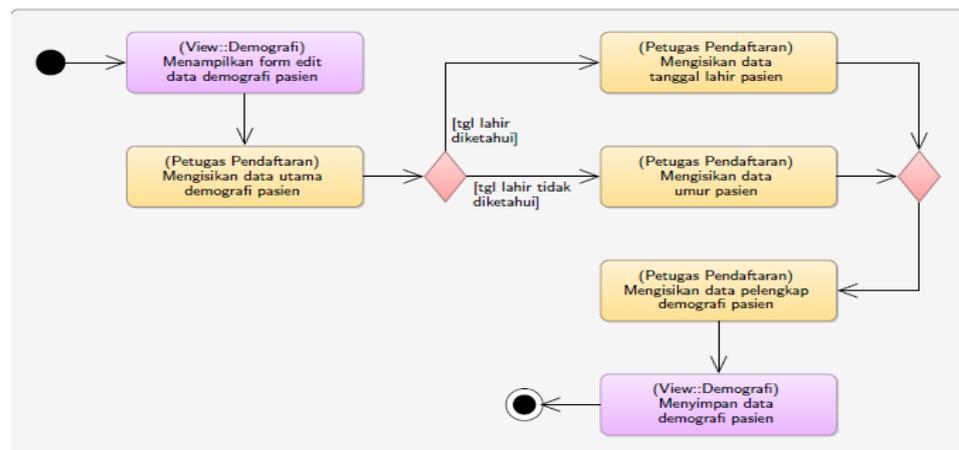
Hasil Data pendaftaran pasien hasil pencarian yang dipilih untuk ditampilkan.

Deskripsi Pencarian data pasien dapat dilakukan berdasarkan nomor rekam medik, nomor antrian, atau berdasarkan nama pasien. Pencarian juga dapat dilakukan secara manual dari daftar data pendaftaran untuk suatu unit layanan tertentu. Pencarian ini hanya

dilakukan pada data pendaftaran dengan tanggal pelayanan hari ini, karena pendaftaran yang bisa dikonfirmasi hanyalah pendaftaran untuk pelayanan hari ini. Alur aktifitas dari kasus penggunaan ini dapat dilihat pada Gambar 3.19. Berdasarkan alur aktifitas tersebut, kasus penggunaan ini dipecah menjadi beberapa kasus penggunaan lagi seperti yang digambarkan pada Gambar 3.20.



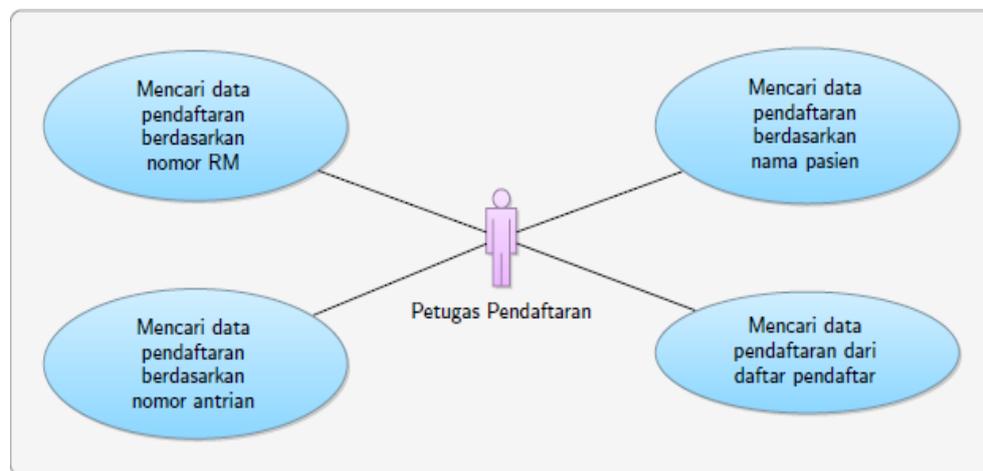
Gambar 3.15: Alur aktifitas edit data demografi pasien



Gambar 3.16: Alur aktifitas mencari data pendaftaran pasien

Ekstensi

- Mencari data pendaftaran berdasarkan nomor RM
- Mencari data pendaftaran berdasarkan nomor antrian
- Mencari data pendaftaran berdasarkan nama pasien
- Mencari data pendaftaran dari daftar pendaftar



Gambar 3.17: Kasus-kasus penggunaan pada mencari data pendaftaran pasien

3.2.2.3 Arsitektur Aplikasi

Dalam pengembangan aplikasi admisi SIMRS PKU Muhammadiyah Wonosobo, arsitektur aplikasi dibagi menjadi 2 bagian, yaitu *Backend Application* dan *Frontend Application*. *Backend Application* berfungsi sebagai data service yang berhubungan langsung dengan database, bagian ini melayani permintaan data dari *Frontend Application*. Sedangkan *Frontend Application* berfungsi sebagai antarmuka user saat menggunakan aplikasi admisi ini.

3.2.2.3.1 Backend Application

Backend Application merupakan aplikasi yang bertujuan untuk melayani permintaan data dari aplikasi *frontend*, aplikasi ini tidak berkomunikasi langsung dengan pengguna (user) melainkan secara langsung berhubungan dengan data base. Backend ini mengadopsi data - data dari web service dan menggunakan tool - tool dari django.

3.2.2.3.1.1 Web Service

Web Service menurut W3.org mendefinisikan sebagai “ sebuah software aplikasi yang dapat teridentifikasi oleh URI dan memiliki *interface* yang didefinisikan,

dideskripsikan, dan dimengerti oleh XML dan juga mendukung interaksi langsung dengan software aplikasi yang lain dengan menggunakan message berbaris XML melalui protokol internet.

Keuntungan Penggunaan Web Service :

- Format penggunaan terbuka untuk semua platform,
- Mudah dimengerti dan mudah men-debug,
- Dukungan interface yang stabil,
- Mudah untuk menangani pesan – pesan proses dan menambahkan nilai,
- Routing dan pengiriman
- Security
- Management and monitoring,
- Schema and service design,
- Akselerasi,
- Mudah untuk dikembangkan dengan semantic transport tambahan,
- Tidak mahal untuk di implementasikan,

- Mengurangi biaya integrasi aplikasi enterprise,
- Banyaknya pengguna web service.

3.2.2.3.1.2 Django

Django adalah sebuah web *framework* yang berbasis python dan menggunakan sistem MTV, yaitu Model, Template dan View. Web *framework* itu sendiri adalah sebuah alat yang digunakan untuk mempermudah kita dalam membangun sebuah website (FEBRIANTO, 2012).

Model adalah layer yang digunakan untuk berinteraksi dengan database, Template adalah layer presentasi untuk HTML, XML dan lainnya, sedangkan View adalah layer yang berisikan logika yang mengolah data dari model dan mengirimkannya ke dalam Template.

Django mempunyai kelebihan diantara web framework lainnya. Beberapa kelebihan yang dimiliki oleh Django adalah:

- **Object-relational mapper:**

Mendefinisikan data model dalam Python dan menggunakan API (Application programming interface) untuk mengakses data tersebut.

- **Automatic admin interface Django:**

Menyediakan admin interface secara otomatis, sehingga anda tidak perlu repot – repot untuk membuatnya.

- **Elegant URL design:**

Pembuatan URL yang lebih mudah dan fleksibel.

- **Template system**

Sistem template Django merupakan salah satu yang bagian yang paling powerful.

- **Internationalization:**

Django telah didesain untuk mempermudah anda dalam membuat web multi bahasa.

3.2.2.3.2 Model Data

Dari analisis yang telah dilakukan, rancangan struktur database aplikasi admisi adalah sebagai berikut.

3.2.2.3.2.1 Model Pendaftaran

- **no_daftar: integer**

Nomor pendaftaran, data ini unik untuk setiap pendaftaran. Dalam database harus diimplementasikan sebagai *integer auto-increment*.

- **demografi: Model::Demografi**

Data demografi dari pasien yang melakukan pendaftaran. Dalam database diimplementasikan sebagai *foreign key*.

- **unit_layanan: Model::UnitLayanan**

Unit layanan yang dituju. Dalam database diimplementasikan sebagai *foreign key*.

- **tgl_daftar: Date**

Tanggal pendaftaran.

- **tgl_layan: Date**

Tanggal pelayanan.

- **ano_antri: integer**

Nomor antrian, setiap pukul 00:00 nomor antrian selalu dimulai dari 1 untuk setiap unit layanan (lihat fungsi `create_and_save()`).

- **keb_rawat: enum**

Kebutuhan perawatan.

Enumerasi

- darurat;
- mendesak;
- segera;
- biasa.

- **jenis_rawat: enum**

Jenis perawatan.

Enumerasi

- rawat inap;
- rawat jalan.

- **jenis_layan: integer**

Jenis layanan.

Enumerasi

- preventif;
- kuratif;
- paliatif;
- rehabilitatif.

- **krit_layan: integer**

Kriteria layanan.

Enumerasi

- biasa;
- intensif;
- khusus.

- **kd_ptgs_daft: integer**

Kode petugas yang melayani pendaftaran.

- **kd_ptgs_confirm: integer**

Kode petugas yang mengkonfirmasi pendaftaran.

- **arriv_time: Datetime**

Waktu kedatangan pasien. Jika data ini tidak null, berarti pendaftaran telah dikonfirmasi.

- **biaya_daftar: Currency**

Biaya pendaftaran.

- **biaya_kartu: Currency**

Biaya pembuatan kartu.

3.2.2.3.2.2 Model Demografi

- **no_rm: string**

Nomor rekam medik (setiap data demografi memiliki nomor rekam medik yang berbeda/unik). Dalam database **bisa saja** diimplementasikan sebagai *primary key*.

- **tgl_reg: Date**

Tanggal registrasi pasien.

- **nama: string**

Nama pasien.

- **nm_ibu: string**

Nama ibu kandung pasien.

- **nm_bapak: string**

Nama bapak kandung pasien.

- **kelamin: enum**

Jenis kelamin pasien.

Enumerasi

- laki-laki;
- perempuan.

- **tmp_lahir: string**

Nama kabupaten/kota tempat kelahiran pasien.

- **tgl_lahir: date**

Tanggal lahir pasien.

- **is_umur: boolean (initial=false)**

Menunjukkan apakah tanggal lahir berdasarkan hasil kalkulasi dari umur pasien, true jika dan hanya jika tanggal lahir adalah hasil kalkulasi.

- **telp: string**

Nomor telepon pasien.

- **darah: enum**

Golongan darah.

Enumerasi

- O;
- A;
- B;
- AB.
- **no_ktp: string**
Nomor KTP pasien.
- **no_sim: string**
Nomor SIM pasien.
- **no_paspor: string**
Nomor paspor pasien.
- **asuransi: string**
Jenis asuransi yang digunakan pasien.
- **no_askes: string**
Nomor asuransi pasien. Data ini harus diisi bila jenis asuransi adalah ASKES.
- **addr_jln: string**
Nama jalan tempat tinggal pasien.
- **addr_no: string**
Nomor rumah pasien.
- **addr_rt: integer**
Nomor RT tempat tinggal pasien.

- **addr_rw: integer**
Nomor RW tempat tinggal pasien.
- **addr_desa: string**
Nama desa/kelurahan tempat tinggal pasien.
- **addr_kec: string**
Nama kecamatan tempat tinggal pasien.
- **addr_kab: string**
Nama kabupaten/kota tempat tinggal pasien.
- **addr_prop: string**
Nama propinsi tempat tinggal pasien.
- **agama: enum**
Agama yang dipeluk pasien.

Enumerasi

- Islam;
 - Kristen;
 - Katolik;
 - Hindu;
 - Budha;
 - Kepercayaan;
 - tidak beragama.
- **pendidikan: enum**

Pendidikan terakhir pasien.

Enumerasi

- tidak sekolah;
 - belum sekolah;
 - TK;
 - SD;
 - SMP;
 - SMA;
 - Akademi/Diploma;
 - S1;
 - S2;
 - S3.
- **pekerjaan: string**

Pekerjaan pasien. Bila pasien bekerja di perusahaan swasta, maka data ini berisi

nama perusahaan tersebut.

Nilai yang disarankan

- tidak bekerja;
- belum bekerja;
- Petani;

- PNS;
- Buruh;
- TNI;
- POLRI;
- Pelajar;
- Mahasiswa;
- Pedagang.

- **stat_nikah: enum**

Status pernikahan pasien.

Enumerasi

- tidak menikah;
 - belum menikah;
 - menikah;
 - duda;
 - janda.
- **nm_pj: string**
Nama penanggung jawab pasien.
 - **pj_telp: string**
Nomor telepon penanggung jawab pasien.
 - **pj_addr_jln: string**
Nama jalan dan nomor rumah penanggung jawab pasien.

- **pj_addr_no: string**
Nomor rumah penanggung jawab pasien.
- **pj_addr_rt: integer**
Nomor RT tempat tinggal penanggung jawab pasien.
- **pj_addr_rw: integer**
Nomor RW tempat tinggal penanggung jawab pasien.
- **pj_addr_desa: string**
Nama desa/kelurahan tempat tinggal penanggung jawab pasien.
- **pj_addr_kec: string**
Nama kecamatan tempat tinggal penanggung jawab pasien.
- **pj_addr_kab: string**
Nama kabupaten/kota tempat tinggal penanggung jawab pasien.
- **pj_addr_prop: string**
Nama propinsi tempat tinggal penanggung jawab pasien.

3.2.2.3.2.3 Model Unit Layanan

- **kd_unit: integer**

Kode dari unit layanan. Dalam database diimplementasikan sebagai *integer auto increment*.

- **nama_unit: integer**

Nama unit layanan.

- **tgl_antrian: Date**

Tanggal saat nomor antrian di-*reset*.

- **next_antrian: integer**

Nomor antrian berikutnya.

3.2.2.3.2.4 Model Konfigurasi

Model ini berisi hanya satu record.

- **biaya_kartu: Currency**

Biaya pembuatan kartu.

- **biaya_daftar: Currency**

Biaya pendaftaran.

3.2.2.3.3 Implementasi

Dari struktur Model Data yang telah dibuat, implementasi dari struktur database aplikasi admisi sebagai berikut.

3.2.2.3.1 Implementasi Data Pendaftaran

```
1 |import datetime
2 |from django.db import models
3 |from django.db import transaction
4 |from django.core.exceptions import ObjectDoesNotExist
5
6 |from demografi import Demografi
7 |from unit_layanan import UnitLayanan
8 |from konfigurasi import Konfigurasi
9
10 |#from kontak import Manufacturer
11
12 |KEB_RAWAT_CHOICE = (
13 |    ("Darurat", "Darurat"),
14 |    ("Mendesak", "Mendesak"),
15 |    ("Segera", "Segera"),
16 |    ("Biasa", "Biasa"),
17 |)
18
19 |JENIS_RAWAT_CHOICE = (
20 |    ("Rawat inap", "Rawat inap"),
21 |    ("Rawat jalan", "Rawat jalan"),
22 |)
23
24 |JENIS_LAYAN_CHOICE = (
25 |    ("Preventif", "Preventif"),
26 |    ("Kuratif", "Kuratif"),
27 |    ("Paliatif", "Paliatif"),
28 |    ("Rehabilitatif", "Rehabilitatif"),
29 |)
30
31 |KRIT_LAYAN_CHOICE = (
32 |    ("Biasa", "Biasa"),
33 |    ("Intensif", "Intensif"),
34 |    ("Khusus", "Khusus"),
35 |)
36 |class Pendaftaran(models.Model):
37 |    demografi = models.ForeignKey('Demografi')
38 |    unit_layanan = models.ForeignKey('UnitLayanan')
39 |    tgl_daftar = models.DateField(auto_now_add=True)
40 |    tgl_layan = models.DateField()
41
42 |    no_antri = models.IntegerField(editable=False)
43
44 |    keb_rawat = models.CharField(max_length=8, choices=KEB_RAWAT_CHOICE)
45 |    jenis_rawat = models.CharField(max_length=11, choices=JENIS_RAWAT_CHOICE)
46 |    jenis_layan = models.CharField(max_length=13, choices=JENIS_LAYAN_CHOICE)
47 |    krit_layan = models.CharField(max_length=13, choices=KRIT_LAYAN_CHOICE)
48 |    arriv_time = models.DateTimeField(null=True, blank=True)
49 |    biaya_daftar = models.DecimalField(max_digits=9, decimal_places=2)
50
51 |    class Meta:
52 |        app_label = 'admission'
53 |        ordering = ['tgl_daftar', 'no_antri']
54
55 |    def save(self, *args, **kwargs):
56 |        if not self.pk:
57 |            # Hey, ini pendaftaran baru... generate no_antri yah
58 |            self.no_antri = Pendaftaran.generate_no_antrian(self.unit_layanan, self.tgl_layan)
59
60 |            # oops jangan lupa biayanya
61 |            self.biaya_daftar = Konfigurasi.get_konfigurasi().biaya_daftar
62 |            #self.biaya_kartu = Konfigurasi.get_konfigurasi().biaya_kartu;
63
64 |            super(Pendaftaran, self).save(*args, **kwargs) # Call the "real" save() method.
65
66 |    @classmethod
67 |    @transaction.commit_manually
68 |    def generate_no_antrian(cls, unit_layanan, tgl_layan):
69 |        # select_for_update will lock the row
70 |        obj = UnitLayanan.objects.select_for_update().get(pk=unit_layanan.pk)
71 |        try:
72 |            # ambil data terakhir dari unit_layanan dan tanggal layanan yang
73 |            # diinginkan
74 |            latest = Pendaftaran.objects.filter(
75 |                unit_layanan=unit_layanan,
76 |                tgl_layan=tgl_layan).latest('id')
77
78 |            # nomor antrian berikutnya adalah nomor antrian dari data `latest` + 1
79 |            next_antrian = latest.no_antri + 1
80 |            print("latest: %d" % latest.id)
81 |        except ObjectDoesNotExist:
82 |            next_antrian = 1
83
84 |        # release the lock
85 |        transaction.rollback()
86 |        return next_antrian
87
```

Gambar 3.18: Implementasi data pendaftaran

3.2.2.3.3.2 Implementasi Data Demografi

```
1 from django.db import models
2 import datetime as dt
3
4
5 KELAMIN = (
6     ("Laki-laki", "Laki-laki"),
7     ("Perempuan", "Perempuan")
8 )
9
10 DARA = (
11     ("A", "A"),
12     ("B", "B"),
13     ("AB", "AB"),
14     ("O", "O")
15 )
16
17 AGAMA = (
18     ("Islam", "Islam"),
19     ("Kristen", "Kristen"),
20     ("Katolik", "Katolik"),
21     ("Hindu", "Hindu"),
22     ("Budha", "Budha"),
23     ("Kepercayaan", "Kepercayaan"),
24     ("Tidak beragama", "Tidak beragama")
25 )
26
27 PENDIDIKAN = (
28     ("Tidak sekolah", "Tidak sekolah"),
29     ("Belum sekolah", "Belum sekolah"),
30     ("TK", "TK"),
31     ("SD", "SD"),
32     ("SMP", "SMP"),
33     ("SMA", "SMA"),
34     ("Akademi/Diploma", "Akademi/Diploma"),
35     ("S1", "S1"),
36     ("S2", "S2"),
37     ("S3", "S3")
38 )
39
40 PEKERJAAN = (
41     ("Tidak bekerja", "Tidak bekerja"),
42     ("Belum bekerja", "Belum bekerja"),
43     ("Petani", "Petani"),
44     ("PNS", "PNS"),
45     ("Guru", "Guru"),
46     ("TNI", "TNI"),
47     ("POLRI", "POLRI"),
48     ("Pelajar", "Pelajar"),
49     ("Mahasiswa", "Mahasiswa"),
50     ("Pedagang", "Pedagang"),
51     ("Swasta", "Swasta")
52 )
53
54 STAT_NIKAH = (
55     ("Tidak menikah", "Tidak menikah"),
56     ("Belum menikah", "Belum menikah"),
57     ("Menikah", "Menikah"),
58     ("Janda", "Janda"),
59     ("Landa", "Landa")
60 )
61
62 class Demografi(models.Model):
63     tgl_reg = models.DateField(auto_now_add=True)
64     nama = models.CharField(max_length=255)
65     no_lha = models.CharField(max_length=20, null=True)
66     no_bapak = models.CharField(max_length=255, null=True)
67     kelamin = models.CharField(max_length=5, choices=KELAMIN)
68     tgl_lahir = models.CharField(max_length=5)
69     tgl_lahir = models.DateField(null=True, blank=True)
70     is_umur = models.BooleanField()
71     tmp = models.CharField(max_length=30)
72     daerah = models.CharField(max_length=5, choices=DARA)
73     no_ktp = models.CharField(max_length=30, null=True)
74     no_sis = models.CharField(max_length=30, null=True)
75     no_paspor = models.CharField(max_length=30, null=True)
76     asuransi = models.CharField(max_length=30, null=True)
77     no_astes = models.CharField(max_length=30, null=True)
78     addr_ln = models.CharField(max_length=50, null=True)
79     addr_no = models.CharField(max_length=5, null=True)
80     addr_rt = models.CharField(max_length=5, null=True)
81     addr_rw = models.CharField(max_length=5, null=True)
82     addr_desa = models.CharField(max_length=30, null=True)
83     addr_kec = models.CharField(max_length=30, null=True)
84     addr_kab = models.CharField(max_length=30, null=True)
85     addr_prov = models.CharField(max_length=30, null=True)
86     agama = models.CharField(max_length=30, choices=AGAMA)
87     pendidikan = models.CharField(max_length=30, choices=PENDIDIKAN)
88     pekerjaan = models.CharField(max_length=30, choices=PEKERJAAN)
89     stat_nikah = models.CharField(max_length=5, choices=STAT_NIKAH)
90     no_pj = models.CharField(max_length=30, null=True)
91     pj_tipe = models.CharField(max_length=30, null=True)
92     pj_addr_ln = models.CharField(max_length=30, null=True)
93     pj_addr_no = models.CharField(max_length=5, null=True)
94     pj_addr_rt = models.CharField(max_length=5, null=True)
95     pj_addr_rw = models.CharField(max_length=5, null=True)
96     pj_addr_desa = models.CharField(max_length=30, null=True)
97     pj_addr_kec = models.CharField(max_length=30, null=True)
98     pj_addr_kab = models.CharField(max_length=30, null=True)
99     pj_addr_prov = models.CharField(max_length=30, null=True)
100
101     class Meta:
102         app_label = "admission"
103         ordering = ['id']
104
105     def save(self, *args, **kwargs):
106         # Jika tanggal lahir berupa umur, buat tanggal lahir sesuai dengan hari ini
107         # no = self.tgl_lahir.strftime("%Y-%m-%d")
108         # if s.find("%") == -1:
109             # Jika tanggal lahir berupa umur, buat tanggal lahir sesuai dengan hari ini
110             # tgl = dt.datetime.now()
111             # self.tgl_lahir = (tgl.year+1900-self.tgl_lahir).strftime("%d/%m/%Y")
112
113         super(Demografi, self).save(*args, **kwargs) # call the "real" save() method.
114
115     def unicode(self):
116         return self.nama
117
118
119
120
121
122
```

Gambar 3.19: Implementasi data pendaftaran

3.2.2.3.3 Implementasi Data Unit Layanan

```
1 from django.db import models
2
3
4 class UnitLayanan(models.Model):
5
6     nama_unit = models.CharField(max_length=50, null=True)
7
8     class Meta:
9         app_label = "admission"
10        ordering = ['id']
11
12
```

Gambar 3.20: Implementasi data Unit Layanan

3.2.2.3.4 Implementasi Data Konfigurasi

```
1 from django.db import models
2
3
4 class Konfigurasi(models.Model):
5     biaya_daftar = models.DecimalField(max_digits=9, decimal_places=2)
6     biaya_kartu = models.DecimalField(max_digits=9, decimal_places=2)
7
8     class Meta:
9         app_label = 'admission'
10
11     @classmethod
12     def get_konfigurasi(cls):
13         return Konfigurasi.objects.get(pk=1)
14
15
```

Gambar 3.21: Implementasi data konfigurasi

BAB IV

KESIMPULAN

4.1 Kesimpulan

Kesimpulan dari Kerja Praktek yang telah dilakukan adalah:

- a. Dihasilkan aplikasi admisi untuk Sistem Informasi Manajemen
- b. Dihasilkan sistem pendaftaran pasien secara komputerisasi
- c. Penyimpanan data pasien menjadi lebih aman
- d. Karyawan Rumah Sakit PKU Muhammadiyah Wonosobo lebih mudah dalam mengelola dan mengakses data pasien.

4.2 Saran

Dari hasil dan pembahasan didapatkan saran sebagai berikut :

- a. Website Rumah Sakit PKU Muhammadiyah Wonosobo ini dapat dikembangkan dengan menambahkan pendaftaran secara *mobile online*.
- b. Menambahkan aksi *insert* pada halaman artikel untuk memasukkan file ke dalam *website back-and*
- c. Membuat *interface* yang lebih *simple* untuk mempermudah user dalam memakainya

d. Meningkatkan kualitas *website* dengan meningkatkan sistem keamanan *website*

DAFTAR PUSTAKA

Barori, A. (2012, April). *dataonline*. Retrieved Mei 24, 2012, from kemenag.co.id: <http://yogyakarta.kemenag.co.id>

Kadir, A. (2003). *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit Andi.

Mulyanto, A. (2009). *Sistem Informasi Konsep & Aplikasi*. Yogyakarta: Pustaka Pelajar.

DAFTAR PUSTAKA

Adiputra, F. (2012). *Sistem Informasi Rumah Sakit Terintegrasi*. YOGYAKARTA.

FEBRIANTO, A. (2012, FEBRUARY 21). Retrieved from

<http://www.makeitfossible.web.id/2012/02/21/pengenalan-django-dan-cara-penginstalannya/>

Putra, G. P. (2012). *Sistem informasi Inventory*. Surabaya: <http://www.slideshare.net>.