**SKPL**-RUMAH SAKIT

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SKPL RUMAH SAKIT

untuk:

Memenuhi Tugas Besar APPL

Dipersiapkan oleh:

1. Jiddy Abdillah (1301162765)

2. Ryan Ramdhani (1301164718)

3. Ari Wilyan Ramadhelza (1301164708)

4. Hendro Pratama Saragih (1301164683)

5. Yudha Febris Farmana (1301164674)

Kelas :

IFIK 40 – 04

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| *SKPL-RUMAH SAKIT* | | 19 |
| Revisi | *-* | *Tgl: 21-09-2018* |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX | - | A | B | C | D | E | F | G |
| TGL |  | 21/09/2018 |  |  |  |  |  |  |
| Ditulis oleh |  | Jiddy  Ryan  Ari  Hendro  Yudha |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

# Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

# Daftar Isi

[Daftar Perubahan 2](#_Toc530977260)

[Daftar Halaman Perubahan 4](#_Toc530977261)

[Daftar Isi 5](#_Toc530977262)

[Daftar Gambar 7](#_Toc530977263)

[Daftar Tabel 7](#_Toc530977264)

[Daftar Lampiran 7](#_Toc530977265)

[1. Pendahuluan 8](#_Toc530977266)

[1.1 Tujuan Penulisan Dokumen 8](#_Toc530977267)

[1.2 Lingkup Masalah 8](#_Toc530977268)

[1.3 Definisi, Singkatan, dan Akronim 8](#_Toc530977269)

[1.4 Referensi 8](#_Toc530977270)

[1.5 Deskripsi Umum Dokumen (Ikhtisar) 9](#_Toc530977271)

[2 Deskripsi Global Perangkat Lunak 9](#_Toc530977272)

[2.1 Perspektif Produk 9](#_Toc530977273)

[2.2 Fungsi Produk 9](#_Toc530977274)

[2.3 Karakteristik Pengguna 10](#_Toc530977275)

[2.4 Batasan-Batasan 11](#_Toc530977310)

[2.5 Asumsi dan Kebergantungan 11](#_Toc530977311)

[2.6 Arsitektur Aplikasi 11](#_Toc530977312)

[3 Deskripsi Rinci Kebutuhan 11](#_Toc530977313)

[3.1 Kebutuhan Antarmuka Eksternal 11](#_Toc530977314)

[3.1.1 Antarmuka pengguna 11](#_Toc530977315)

[3.1.2 Antarmuka perangkat keras 12](#_Toc530977316)

[3.1.3 Antarmuka perangkat lunak 12](#_Toc530977317)

[3.1.4 Antarmuka komunikasi 12](#_Toc530977318)

[3.2 Kebutuhan Fungsional 12](#_Toc530977319)

[3.2.1 Use Case Diagram 12](#_Toc530977320)

[3.2.2 Use Case Scenario 13](#_Toc530977321)

[3.3 Deskripsi Data 16](#_Toc530977322)

[3.3.1 Kamus data 17](#_Toc530977323)

[3.4 Deskripsi Kebutuhan Non Fungsional 18](#_Toc530977324)

[3.4.1 Availability 18](#_Toc530977325)

[3.4.2 Access 18](#_Toc530977326)

[3.4.3 Recovery 18](#_Toc530977327)

[3.5 Atribut Kualitas Perangkat Lunak 18](#_Toc530977328)

[3.6 Batasan Perancangan 18](#_Toc530977329)

[3.7 Matriks Keterunutan 19](#_Toc530977330)

[Lampiran 19](#_Toc530977331)

[Flow Map/Prosedur 19](#_Toc530977332)

[Lampiran lain yang dianggap perlu 19](#_Toc530977333)

# Daftar Gambar

Hanya dicantumkan dan diisi jika ada gambar pada badan dokumen

# Daftar Tabel

Hanya dicantumkan dan diisi jika ada tabel pada badan dokumen

# Daftar Lampiran

Hanya dicantumkan dan diisi jika ada lampiran setelah badan dokumen

# 1. Pendahuluan

Dokumen ini berisi spesifikasi kegiatan yang dilakukan di rumah sakit, mulai dari pasien melakukan pendaftaran, pasien diperiksa oleh dokter yang menangani, hingga pasien selesasi melakukan administrasi rumah sakit. Untuk penamaan dokumen ini selanjutnya akan digunakan istilah SKPL.

## Tujuan Penulisan Dokumen

Dokumen spesifikasi kegiatan yang dilakukan dirumah sakit merupakan spesifikasi kebutuhan masyarakat (pasien) akan pentingnya penanganan di bidang kesehatan. Dokumen ini digunakan oleh pihak rumah sakit sebagai acuan teknis standar operasional penanganan kesehatan di rumah sakit.

## Lingkup Masalah

SKPL Rumah Sakit ini merupakan sistem informasi yang berbasis web untuk monitoring atau memantau kegiatan yang berjalan di rumah sakit agar dalam pelaksanaannya tidak ada kesalah pahaman antara pasien dengan dokter atau pun pihak rumah sakit.

## Definisi, Singkatan, dan Akronim

* SKPL adalah Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggris-nya sering juga disebut sebagai software requirements specification (SRS), dan merupakan spesifikasi perangkat lunak yang akan dikembangkan.
* SKPL-RS-xxx adalah kode yang digunakn untuk merepresentasikan kebutuhan (requirement) pada SKPL Rumah Sakit, dengan merupakan kode perangkat lunak, RSSKPL adalah kodefase, dan xxx adalah digit/nomor penggunaan
* Use Case Diagram adalah gambaran graphical dari beberapa atau semua actor, **use case**, dan interaksi diantaranya yang memperkenalkan suatu sistem.
* Use Case Scenario adalah sebuah narasi untuk menjelaskan use case yang ada sehingga customer lebih mudah memahami fungsi dari use case yang ada.
* Class Diagram adalah model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class.
* ERD adalah entity relationship diagram, diagram dan notasi yang digunakan untuk mempresentasikan struktur data statis pada perangkat lunak.
* UML adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya.
* HTML adalah Hyper Text Markup Language, sintaks bahas yang digunakan dalam world wide web.
* JavaScript adalah bahasa skrip yang populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera. Kode JavaScript dapat disisipkan dalam halaman web menggunakan tag SCRIPT.
* CSS (Cascading Style Sheet) Merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.
* PHP (Hypertext Preprocessor) Adalah bahasa skrip yang dapat ditanamkan atau disisipkan kedalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis.

## Referensi

* <https://hanungnp.staff.telkomuniversity.ac.id/files/2015/04/contoh-SKPL-Sistem-informasi-tugas-akhir-SISTA.pdf>
* <http://whatyoun.blogspot.com/2015/12/contoh-skpl-rekayasa-perangkat-lunak.html>
* <http://www.academia.edu/11355568/Contoh_SKPL_SPESIFIKASI_KEBUTUHAN_PERANGKAT_LUNAK_>
* <http://ieeechicago.org/skpl-events/>
* https://www.codepolitan.com/mengenal-uml-diagram-use-case

## Deskripsi Umum Dokumen (Ikhtisar)

Dokumen SKPL ini dibagi menjadi tiga bagian utama. Bagian utama berisi penjelasan tentang dokumen skpl yang mencakup tujuan pembuatan dokuman ini, lingkup masalah yang diselesaikan oleh perangkat lunak yang dikembangkan definisi, referensi, dan deskripsi umum. Bagian kedua berisi penjelasan secara umum mengenai perangkat lunak yang akan dikembangkan meliputi fungsi dari perangkat lunak, karakteristik pngguna, batasan dan asumsi yang diambil dalam pengembangan perangkat lunak. Bagian ketiga berisi uraian kebutuhan perangkat lunak secara lebih rinci.

# Deskripsi Global Perangkat Lunak

Perangkat lunak sistem informasi rumah sakit ini dibuat sebagai panduan untuk pasien guna mempermudah proses pendaftaran, pemilihan dokter, dan juga pembayaran. Selain itu, manfaat lain yang dapat dirasakan untuk dokter yaitu untuk mempermudah sinkronisasi jadwal antara dokter dengan pasien. Lalu, user lain yang berhak untuk mengakses sistem informasi rumah sakit ini ialah pihak rumah sakit (*administrator)* guna mempermudah proses update dan juga penghapusan data pasien yang tidak sesuai ataupun yang sudah tidak berlaku. Pada perangkat lunak ini user dibagi menjadi tiga bagian, seperti yang sudah disebutkan diatas. Ketiga nya memiliki hak akses yang berbeda, namun bisa sama-sama menggunakan sistem informasi ini sesuai dengan aturan masing-masing.

## Perspektif Produk

RSSKPL adalah perangkat lunak Sistem Informasi Rumah Sakit berbasis web untuk monitoring atau memantau kegiatan yang berlangsung di rumah sakit tersebut, dimulai dari pendaftaran pasien penanganan dokter terhadap pasien, penempatan kamar inap pada pasien, pemberian obat-obatan terhadap pasien yang telah dibuat oleh dokter yang menanganinya. Setelah proses pemulihan pasien, pasien wajib membayar biaya administrasi yang telah di setujui sebelumnya. Sehingga dalam pelaksanaanya pasien menjadi lebih terarah dan teratur sehingga dapat mempercepat masa penyembuhan.

## Fungsi Produk

Adapun fungsi-fungsi yang dimiliki oleh perangkat lunak ini adalah:

1. Pasien

* Registrasi untuk pasien yang baru pertama kali melakukan pendaftaran.
* Melihat jadwal dokter yang tersedia di rumah sakit.
* Memilih jadwal periksa yang sesuai dengan jadwal dokter.

2. Dokter

* Melihat list pasien yang ingin ditanganinya.
* Melihat dan mengupdate jadwal praktek.
* Mengirim resep obat ke apoteker.
* Mengirim invoice ke resepsionis.

3. Apoteker

* Melihat resep yang dikirim oleh dokter.
* Mengupdate stok obat.
* Melihat data obat.

4. Perawat

* Mengisi data perawat dan kamar pasien yang dirawat.
* Melihat dan mengupdate ketersediaan kamar.

5. Resepsionis

* Melihat jadwal periksa pasien.
* Melihat jadwal praktek dokter.
* Melihat data rawat inap pasien.
* Menginput data pembayaran.

## Karakteristik Pengguna

Pengguna sistem informasi ini adalah pasien, dokter, apoteker, perawat dan resepsionis di Rumah Sakit yang sudah menerapkan sistem ini. Dokter adalah yang berperan sebagai *eksekutor* dalam penanganan medis yang dibantu oleh Bagian Administrasi di rumah sakit tersebut. Pasien adalah orang yang hendak melakukan pemeriksaan. Apoteker berperan sebagai peracik dan menangani masalah obat-obatan. Perawat mempunyai peran yaitu merawat pasien. Yang terakhir, resepsionis berperan sebagai pelayan terhadap customer yang ada di rumah sakit.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pengguna | Tanggung Jawab | Hak Akses | Fitur | Status | Ket |
| Pasien | Melakukan registrasi sebelum menggunakan pelayanan rumah sakit | Pasien dapat mengakses jadwal dokter yang ada | - Registrasi - Login  - Melihat jadwal praktek dokter  - Memilih jadwal periksa | -Aktif -Non-aktif | - |
| Dokter | Menyediakan jadwal praktek dan memberi resep ke apoteker | Memiliki hak akses data pasien dan data dokter itu sendiri. | - Login - Melihat dan mengupdate jadwal jaga praktik  - Melihat pasien yang ditangani  - Mengirim resep ke apoteker | - Aktif | - |
| Apoteker | Meracik resep dan mengontrol obat | Dapat mengakses data obat | - Login - Melihat resep obat  - Mengupdate stok obat | - Aktif | - |
| Perawat | Mengisi data rawat inap dan mengontrol ketersediaan kamar | Perawat dapat mengakses data pasien, perawat, dan kamar | - Login - Mengisi data rawat inap  - Melihat dan mengupdate ketersediaan kamar | - Aktif |  |
| Resepsionis | Melakukan pelayan terhadap customer | Mengakses data pasien, dokter, kamar | - Login - Melihat jadwal periksa pasien  - Melihat jadwal praktek dokter  - Melihat data rawat inap pasien | - Aktif |  |

## Batasan-Batasan

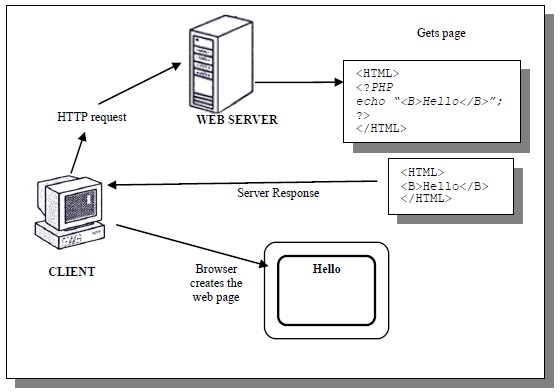
Batasan-batasan yang digunakan pada pengembangan perangkat lunak ini adalah:

* Menggunakan jaringan area local di rumah sakit tersebut.
* Hanya pasien dan pegawai rumah sakit yang bisa mengakses sistem informasi ini.
* Untuk mengakses sistem informasi harus dilakukan login terlebih dahulu.

## Asumsi dan Kebergantungan

Perangkast lunak ini hanya dapat digunakan didalam jaringan area local. Bagi pengguna user pasien, pasien hanya dapat mengakses sistem informasi ini di rumah sakit tertentu yang sudah menerapkan sistem informasi ini.

## Arsitektur Aplikasi

Arsitektur aplikasi merupakan teknologi yang digunakan untuk mengimplementasikan satu atau lebih beberapa dari sistem informasi yang ada dalam sisfors rumah sakit ini. Arsitektur aplikasi ini berfungsi sebagai outline untuk desain, konstruksi dan implementasi.

# Deskripsi Rinci Kebutuhan

Dalam BAB ini akan dijelaskan kebutuhan apa saja yang akan digunakan dalam pembentukan sistem informasi rumah sakit (SISFORS).

## Kebutuhan Antarmuka Eksternal

Kebutuhan antarmuka eksternal pada perangkat lunak SISFORS mencakup kebutuhan antar muka pemakai, dan antar muka perangkat lunak.

### Antarmuka pengguna

Antarmuka pemakai akan dikembangkan menggunakan modus klik’n’go dan berbasis web. Pemakai berinteraksi dengan perangkat lunak SISFORS melalui antarmuka penjelajah situs (web browser) yang disediakan oleh rumah sakit. SISFORS menerima masukan dari pemakai melalui perintah yang diklik pada mouse atau yang diketikkan melalui keyboard seperti operan pertama, operan kedua, atau tipe operator yang sesuai dengan operasi aritmatika yang diinginkan. Keluaran dari perangkat lunak PELITIKA dapat dilihat pemakai dengan menggunakan monitor secara langsung.

### Antarmuka perangkat keras

Kebutuhan minimum perangkat keras yang dapat digunakan oleh SISFORS adalah :

* PC Standard
* *Keyboard*
* *Mouse*

### Antarmuka perangkat lunak

Untuk mengakses SISFORS hanya menggunakan browser explorer atau Microsft Edge. Jadi tidak diizinkan menggunakan browser diluar aturan tersebut.

### Antarmuka komunikasi

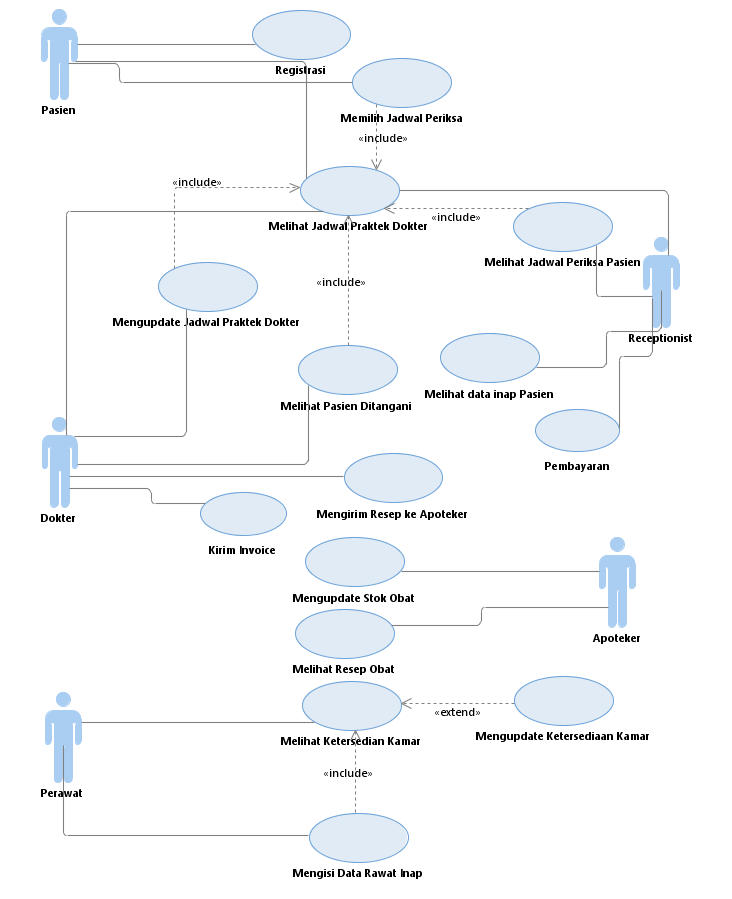
Yang dibutuhkan hanya sebuah computer server dan satu atau beberapa computer sebagai client yang terhubung secara client-server dalam lingkup jaringan area local atau internet berbasis TCP/IP.

## Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan aksi dasar pada perangkat lunak ini, aksi dasar yang dilakukan perangkat lunak ini adalah menyediakan informasi mengenai kegiatan pelaksanaan aktivitas yang berlangsung di rumah sakit. Di setiap proses pada perangkat lunak ini memiliki tahapan yang berbeda dan juga keluaran yang berbeda.

### Use Case Diagram

Pada Use Case Diagram ini, terdapat 5 aktor, yaitu pasien, dokter, resepsionis, apoteker, dan perawat. Setiap actor memiliki use case/ proses masing-masing. Ada satu use case yang memiliki banyak actor, yaitu use case melihat jadwal praktek dokter.



### Use Case Scenario

#### Registrasi

Use case : Registrasi

Actor : Pasien

Pre-condition : Pasien datang ke resepsionis

Post-condition : Pasien mendapatkan username dan password untuk login ke sisfoRS

Deskripsi :

* Pasien masuk ke menu utama sisfoRS
* Pasien memilih menu registrasi
* Pasien mengisi data diri secara lengkap dan benar sesuai dengan yang diminta pada bagian registrasi sisfoRS

#### Lihat Jadwal Praktek Dokter

Use case : Melihat Jadwal Praktek Dokter

Actor : Pasien, Dokter, Resepsionis

Pre-condition : 1. Aktor telah login ke sisfoRS

2. Dokter telah menginput jadwal praktek

Post-condition : Aktor dapat melihat jadwal praktek dokter

Deskripsi :

* Aktor masuk ke menu utama sisfoRS
* Aktor memilih menu lihat jadwal praktek dokter
* Jadwal praktek dokter ditampilkan

#### Memilih Jadwal Periksa

Use case : Memilih Jadwal Periksa

Actor : Pasien

Pre-condition : 1. Pasien telah login ke sisfoRS

2. Dokter telah menginput jadwal praktek

Post-condition : Pasien mendapatkan jadwal periksa

Deskripsi :

* Pasien masuk ke menu utama sisfoRS
* Pasien masuk ke menu pilih jadwal periksa
* Jadwal dokter ditampilkan
* Pasien mengisi jadwal periksa sesuai dengan jadwal yang ada

#### Update Jadwal Praktek Dokter

Use case : Update Jadwal Praktek Dokter

Actor : Dokter

Pre-condition : Dokter telah login ke sisfoRS

Post-condition : Jadwal praktek dokter telah ter-update

Deskripsi :

* Dokter masuk ke menu utama sisfoRS
* Dokter masuk ke menu update jadwal praktek
* Dokter bisa mengupdate atau menambah jadwal praktek

#### Lihat Pasien yang Ditangani

Usecase : Melihat Pasien yang Ditangani

Actor : Dokter

Pre-condition : 1. Dokter telah login ke sisfoRS

2. Dokter telah mengisi jadwal praktek

Post-condition : Dokter telah melihat pasien yang akan ditangani sesuai jadwal yang dipilih

Deskripsi :

* Dokter masuk ke menu utama sisfoRS
* Dokter masuk ke menu pasien yang ditangani
* Jadwal dokter ditampilkan
* Dokter memilih jadwal untuk melihat pasien yang akan ditangani sesuai dengan jadwal

#### Kirim Resep ke Apoteker

Usecase : Mengirim Resep ke Apoteker

Actor : Dokter

Pre-condition : 1. Dokter telah login ke sisfoRS

2. Dokter telah memeriksa pasien

Post-condition : Apoteker menerima resep dari dokter

Deskripsi :

* Dokter masuk ke menu utama sisfoRS
* Dokter masuk ke menu kirim resep
* Dokter meinginputkan resep sesuai pasien yang diperiksa
* Dokter mengirim resep obat pasien ke apoteker

#### Lihat Resep Obat

Usecase : Melihat Resep Obat

Actor : Apoteker

Pre-condition : 1. Apoteker telah login ke sisfoRS

2. Dokter sudah mengirim resep ke apoteker

Post-condition : Dokter sudah mengirim resep obat pasien ke apoteker

Deskripsi :

* Apoteker masuk ke menu utama sisfoRS
* Apoteker masuk ke menu lihat resep
* Apoteker melihat resep yang sudah dikirim

#### Update Stok Obat

Usecase : Mengupdate Stok Obat

Actor : Apoteker

Pre-condition : Apoteker telah login ke sisfoRS

Post-condition : Stok obat telah terupdate

Deskripsi :

* Apoteker masuk ke menu utama sisfoRS
* Apoteker masuk ke menu update stok obat
* List obat ditampilkan
* Apoteker dapat memilih obat yang akan diupdate stoknya

#### Isi Data Rawat Inap

Usecase : Mengisi Data Rawat Inap

Actor : Perawat

Pre-condition : 1. Perawat telah login ke sisfoRS

2. Pasien sudah diperiksa oleh dokter

3. Kamar inap tersedia.

Post-condition : Data rawat inap telah tersimpan

Deskripsi :

* Perawat masuk ke menu sisfoRS
* Perawat masuk ke menu isi data rawat inap
* List kamar dan ketersediaannya ditampilkan
* Perawat mengisi data rawat inap pasien sesuai dengan ketersediaan kamar

#### Lihat dan Update Ketersediaan Kamar

Usecase : Melihat dan Mengupdate Ketersediaan Kamar

Actor : Perawat

Pre-condition : Perawat sudah login ke sisfoRS

Post-condition : Ketersediaan kamar telah terupdate

Deskripsi :

* Perawat masuk ke menu sisfoRS
* Perawat masuk ke menu lihat ketersediaan kamar
* List kamar ditampilkan
* Perawat memilih kamar yang ingin dilihat ketersediaannya
* Jika perawat ingin mengupdate ketersediaan kamar, perawat pilih menu update ketersediaan kamar
* Perawat mengupdate ketersediaan kamar

#### Lihat Jadwal Periksa Pasien

Usecase : Melihat Jadwal Periksa Pasien

Actor : Resepsionis

Pre-condition : 1. Resepsionis telah login ke sisfoRS

2. Pasien telah memilih jadwal periksa

Post-condition : Resepsionis telah melihat jadwal periksa pasien

Dekripsi :

* Resepsionis masuk ke menu sisfoRS
* Resepsionis masuk ke menu jadwal
* Resepsionis masuk ke menu jadwal periksa pasien
* Resepsionis melihat jadwal periksa pasien

#### Lihat Data Rawat Inap Pasien

Usecase : Melihat Data Rawat Inap Pasien

Actor : Resepsionis

Pre-condition : 1. Resepsionis telah login ke sisfoRS

2. Perawat telah menginput data rawat inap pasien

Post-condition : Resepsionis melihat data rawat inap pasien

Dekripsi :

* Resepsionis masuk ke menu sisfoRS
* Resepsionis masuk ke menu data
* Resepsionis masuk ke menu data rawat inap
* Resepsionis melihat data rawat inap pasien

#### Input Pembayaran

Use case : Menginput Pembayaran

Actor : Resepsionis

Pre-condition : 1. Resepsionis telah login ke sisfoRS

2. Invoice sudah dikirm oleh dokter

Post-condition : Proses pembayaran selesai dan data pembayaran terinput di database

Dekripsi :

* Resepsionis masuk ke menu sisfoRS
* Resepsionis masuk ke menu Input Pembayaran
* List Rekap Pembayaran ditampilkan
* Pilih Rekap Pembayaran
* Data pembayaran diinput ke database

#### Kirim Rekap Pembayaran

Use case : Mengirim Rekap Pembayaran

Actor : Dokter

Pre-condition : 1. Dokter login ke sisfoRS

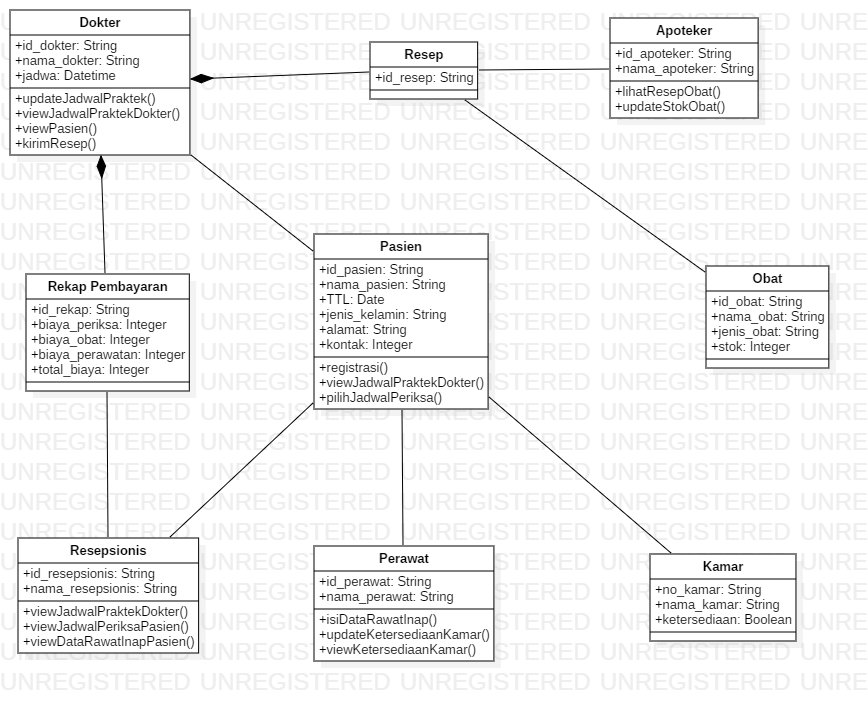
2. Pasien sudah diperiksa

Post-condition : Proses pembayaran selesai dan data pembayaran terinput di database

Dekripsi :

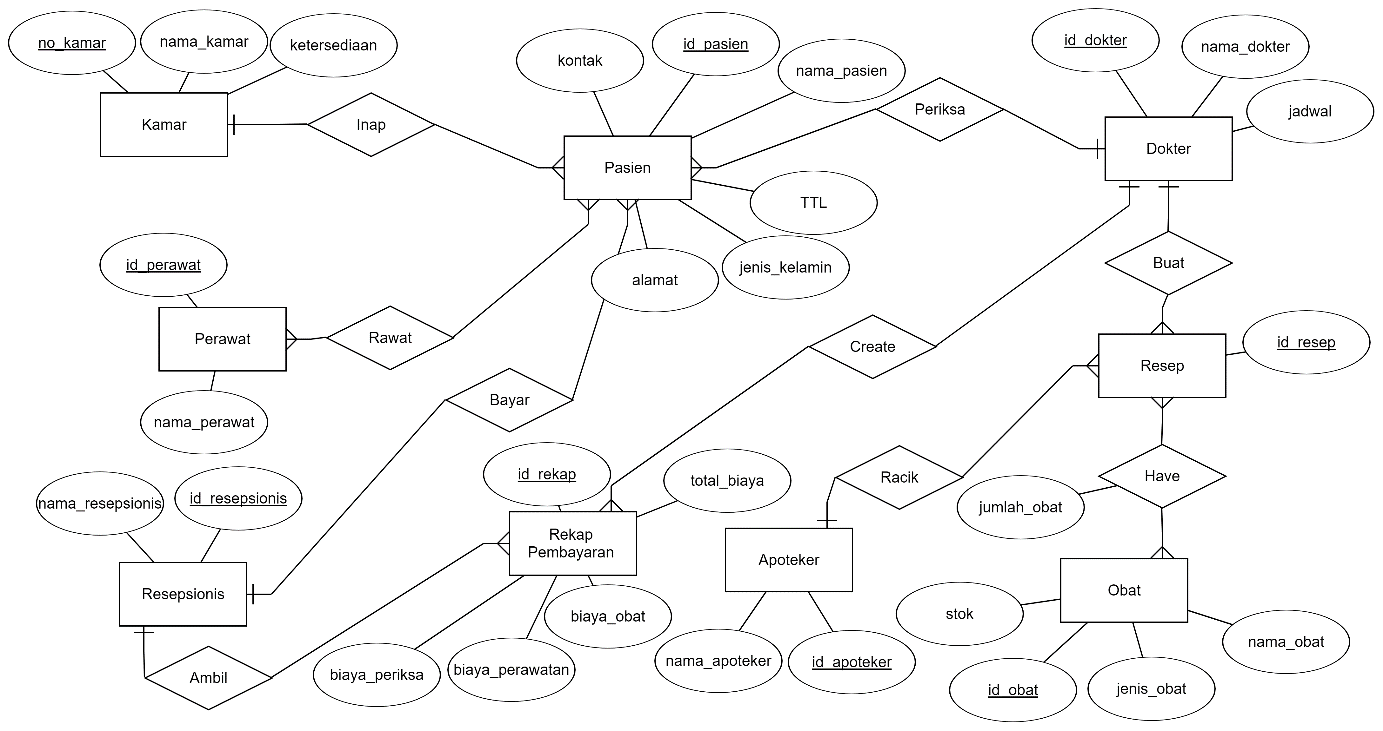
* Dokter masuk ke menu sisfoRS
* Dokter masuk ke menu Kirim Rekap Pembayaran
* Input Rekap Pembayaran
* Mengirim rekap pembayaran ke resepsionis

### Class Diagram



## Deskripsi Data

ERD (Entity Relationship Diagram)



### Kamus data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nama Data | Letak | Atribut | Tipe Data | Key |
| Entitas Dokter |  | id\_dokter  nama\_dokter  jadwal | String  String  Datetime | PK |
| Entitas Pasien |  | id\_pasien  nama\_pasien  alamat  TTL  Kontak  jenis\_kelamin | String  String  String  Date  Integer  String | PK |
| Entitas Obat |  | id\_obat  nama\_obat  jenis\_obat  stok | String  String  String  Integer | PK |
| Entitas Resep |  | id\_resep | String | PK |
| Entitas Kamar |  | no\_kamar  nama\_kamar  ketersediaan | String  String  Boolean | PK |
| Entitas Perawat |  | id\_perawat  nama\_perawat | String  String | PK |
| Entitas Resepsionis |  | id\_resepsionis  nama\_resepsionis | String  String | PK |
| Entitas Apoteker |  | id\_apoteker  nama\_apoteker | String  String | PK |
| Relasi Obat-Resep |  | id\_resep  id\_obat  jumlah\_obat | String  String  Integer | FK  FK |
| Relasi Perawat-Pasien |  | id\_perawat  id\_pasien | String  String | FK  FK |
| Entitas Invoice |  | id\_invoice  biaya\_periksa  biaya\_obat  biaya\_perawatan  total\_biaya | String  Integer  Integer  Integer  Integer | PK |

## Deskripsi Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional berfungsi untuk spesifikasi pendukung dari sistem, dan mengevaluasi operasi dari sebuah sistem, bukan operasi atau proses spesifik. Kebutuhan Non-Fungsional dari sistem informasi rumah sakit ini yaitu:

### Availability

Karena jam operasi rumah sakit harus 24 jam, maka sistem informasi ini dibutuhkan server yang beroperasi 24 jam juga. Tetapi karena user yang mengakses tidak terlalu banyak, maka tidak perlu menggunakan server dengan spesifikasi yang tinggi.

### Access

Setiap actor di sistem informasi ini mempunyai hak akses yang berbeda-beda, dan juga memiliki beberapa akses yang sama, diantara akses tersebut yaitu :

|  |  |
| --- | --- |
| **Actor** | **Akses** |
| Pasien | Pasien dapat mengakses jadwal dokter yang ada |
| Dokter | Memiliki hak akses data pasien dan data dokter itu sendiri. |
| Apoteker | Dapat mengakses data obat |
| Perawat | Perawat dapat mengakses data pasien, perawat, dan kamar |
| Resepsionis | Mengakses data pasien, dokter, kamar |

### Recovery

Untuk menghindari data loss yang besar ketika ada kesalahan pada server, maka semua data yang sudah diinput ke database dibackup setiap hari ke server yang lokasinya berbeda.

## Atribut Kualitas Perangkat Lunak

|  |  |
| --- | --- |
| **Attribut Tipe** | **Keterangan Kualitas Perangkat Lunak** |
| *Performance* | Mempercepat proses unggah maupun unduh dalam akses ke menu dalam fitur. |
| *Bug-Fixed* | Menangani setiap bug yang ada dalam setiap perubahan versi pada perangkat lunak. |
| *Maintenance* | Proses pemeliharaan di perangkat lunak, dilakukan secara rutin maupun dalam keadaan darurat, situasional bergantung pada tingkat *crash.* |
| *Secured-Safe* | Kemanan perangkat lunak yang ditanamkan dalam sistem, sehingga data user terjamin tidak akan ada data yang disebarluaskan terhadap public maupun pihak ketiga. |

## Batasan Perancangan

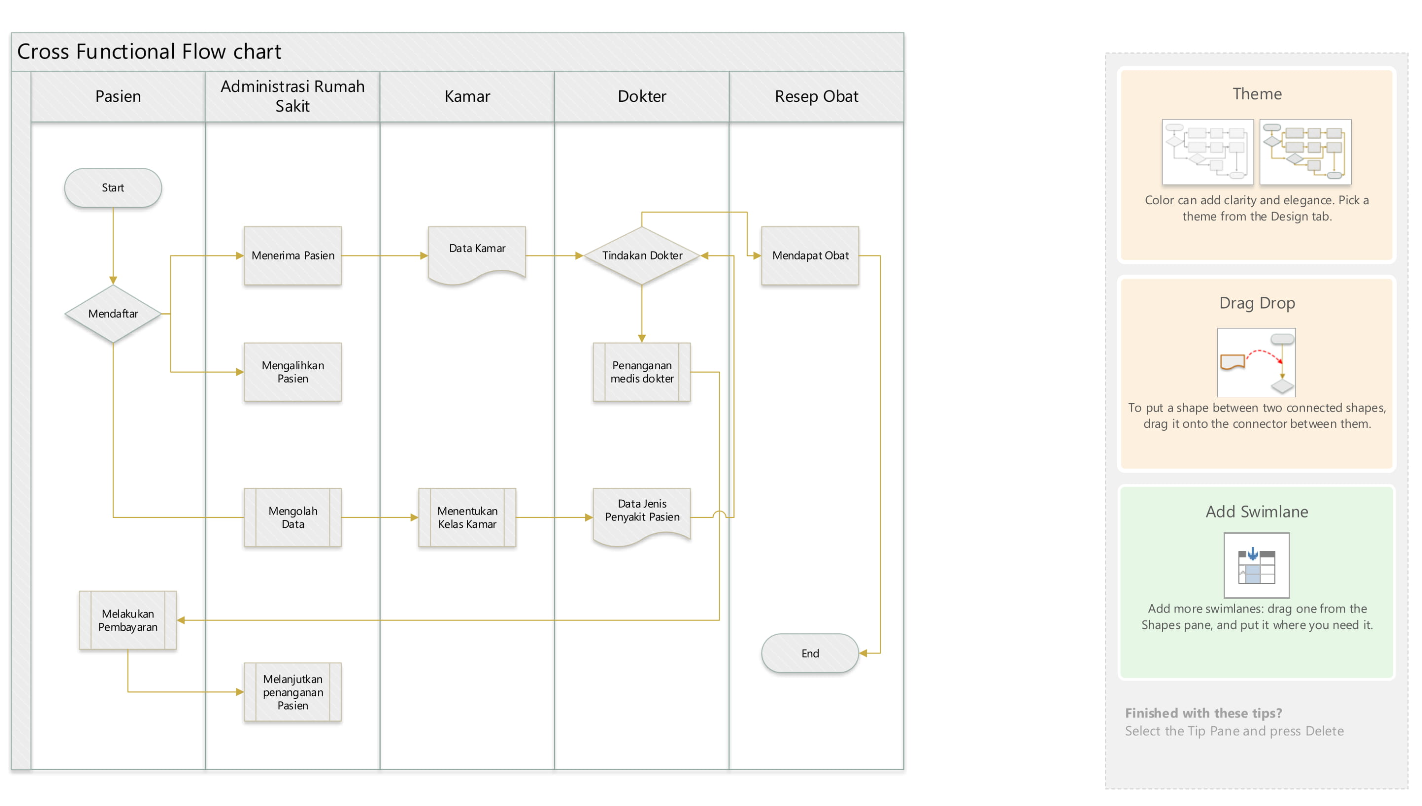
Batasan-batasan yang digunakan pada pengembangan perangkat lunak ini adalah:

* Menggunakan jaringan area local di rumah sakit tersebut.
* Hanya pasien dan pegawai rumah sakit yang bisa mengakses sistem informasi ini.
* Untuk mengakses sistem informasi harus dilakukan login terlebih dahulu.

## Matriks Keterunutan

## Lampiran

### Flow Map/Prosedur

* FlowMap

### Lampiran lain yang dianggap perlu