



PERANCANGAN ANTARMUKA SISTEM INFORMASI SENSUS HARIAN RAWAT INAP MENGGUNAKAN METODE WATERFALL

Hanifah Cahyani¹⁾, Desfa Anisa²⁾

^{1,2} D3 Rekam Medis dan Informasi Kesehatan, Universitas Awal Bros

³ Program Studi Sistem Informasi, Institut Teknologi dan Bisnis Indobaru Nasional

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 30 Desember 2025

Revisi Akhir: 26 Maret 2026

Diterbitkan Online: 30 Maret 2026

Keywords

Daily Inpatient Census, Waterfall, Information System.

KORESPONDENSI

E-mail: hannichayani@gmail.com

ABSTRACT

Manual management of the Daily Inpatient Census (SHRI) in hospitals often faces obstacles such as incomplete data entry and the risk of document loss.

This study aims to design a web-based SHRI information system interface to improve the efficiency and accuracy of patient data management. The research method used is the Waterfall method, which includes the stages of needs analysis, system design, and testing. System development was carried out using the PHP programming language with the CodeIgniter 4 framework and a MySQL database. The results of the study indicate that the system is able to integrate incoming, transferred, and discharged patient data automatically, thereby minimizing manual input errors. Based on testing using the Black Box Testing method, all functional modules of the system were declared to run successfully and in accordance with user needs. Overall, this system design provides an effective digital solution and simplifies the daily census recapitulation reporting process for hospital management.

ABSTRAK

Pengelolaan Sensus Harian Rawat Inap (SHRI) secara manual di rumah sakit sering kali menghadapi kendala berupa ketidaklengkapan pengisian data dan risiko kehilangan dokumen. Penelitian ini bertujuan untuk merancang desain antarmuka sistem informasi SHRI berbasis web guna meningkatkan efisiensi dan akurasi pengelolaan data pasien. Metode penelitian yang digunakan adalah *Waterfall*, yang mencakup tahapan analisis kebutuhan, desain sistem, hingga pengujian. Pengembangan sistem dilakukan menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter 4 dan basis data MySQL. Hasil penelitian menunjukkan bahwa sistem yang dibangun mampu mengintegrasikan data pasien masuk, pindahan, dan keluar secara otomatis, sehingga meminimalkan kesalahan input manual. Berdasarkan pengujian menggunakan metode *Black Box Testing*, seluruh modul fungsional sistem dinyatakan berjalan dengan sukses dan sesuai dengan kebutuhan pengguna. Secara keseluruhan, perancangan sistem ini memberikan solusi digital yang efektif dan mempermudah proses pelaporan rekapitulasi sensus harian bagi pihak manajemen rumah sakit.

Kata Kunci: Sensus Harian Rawat Inap, Waterfall, Sistem Informasi.

PENDAHULUAN

Rumah sakit adalah institusi kesehatan yang berfungsi untuk menyelenggarakan pelayanan kesehatan berupa pelayanan rawat jalan, rawat inap, pelayanan gawat darurat, pelayanan rujukan yang mencakup pelayanan rekam medis dan penunjang medis yang dimanfaatkan untuk penelitian, pendidikan dan pelatihan tenaga kesehatan [1]. Sebagai institusi pelayanan kesehatan, Rumah Sakit sangat bergantung pada teknologi informasi guna mengoptimalkan kinerja dalam memberikan pelayanan kepada pasien. Teknologi informasi merupakan sinergi antara sistem komputer untuk manajemen data dan teknologi komunikasi yang memungkinkan penyebaran informasi secara efisien.

Salah satu upaya yang dilakukan oleh Rumah Sakit dalam proses penyembuhan dan pemulihan pasien adalah penyelenggaraan pelayanan rawat inap. Pelayanan rawat inap memiliki pengaruh yang signifikan terhadap pendapatan Rumah Sakit. Salah satu aspek penting yang harus diperhatikan dalam pelayanan rawat inap adalah sensus harian rawat inap. Sensus harian pasien rawat inap memuat informasi mengenai seluruh pasien yang masuk, pindah ruangan, dipindahkan, serta pasien yang keluar, baik dalam kondisi hidup maupun meninggal dunia, dalam kurun waktu 24 jam setiap hari. Setiap bangsal wajib mengisi lembar sensus harian secara rutin dan mengirimkannya ke unit rekam medis untuk diproses menjadi informasi kesehatan. Selanjutnya, data sensus harian rawat inap tersebut direkapitulasi dan diolah sehingga menghasilkan informasi yang diperlukan [2].

Setelah melakukan observasi di lapangan dan wawancara pada petugas *analyzing reporting* di Rumah Sakit X, pengisian Sensus Harian Rawat Inap (SHRI) saat ini sudah menggunakan sistem pengisian Sensus Harian Rawat Inap berbasis elektronik. Pengiriman SHRI dilakukan setiap hari oleh petugas ruangan rawat inap melalui link dari petugas Rekam Medis. Tetapi pemanfaatan data sensus harian tidak dilakukan secara maksimal dikarenakan pengisian form SHRI oleh petugas ruangan rawat inap yang tidak lengkap. Hal tersebut menjadi kendala dalam pembuatan indikator penggunaan tempat tidur salah satunya karena ketidaklengkapan pengisian SHRI, seperti data pasien yang pindah dari ruangan A ke ruangan B tidak diinput sebagai data pasien pindahan pada form SHRI sehingga petugas pelaporan kesulitan dalam penghitungan indikator tempat tidur.

Berdasarkan latar belakang tersebut, peneliti tertarik untuk melakukan perancangan desain antarmuka sistem Sensus Harian Rawat Inap. Perancangan ini merupakan salah satu tahapan penting sebagai dasar pengembangan sistem Sensus Harian Rawat Inap di masa mendatang, serta memiliki peran yang sangat besar dalam mendukung kemudahan dan kenyamanan pengguna dalam mengoperasikan sistem tersebut secara optimal.

a. Sistem Informasi

Sistem informasi merupakan salah satu komponen penting dalam dunia modern yang berperan sebagai penghubung antara teknologi, manusia, dan proses bisnis dalam suatu organisasi. Secara umum, sistem informasi dapat didefinisikan sebagai

sekumpulan komponen yang saling berinteraksi untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi yang dibutuhkan dalam mendukung kegiatan operasional, manajerial, maupun pengambilan keputusan di suatu entitas [3].

b. Sensus Harian Rawat Inap

Sensus harian rawat inap adalah suatu aktivitas rutin yang dilakukan secara langsung menghitung jumlah pasien yang dilayani di unit rawat inap. Data sensus merupakan data yang harus dikumpulkan setiap hari dan merupakan aktivitas pasien selama 24 jam periode waktu lapor. Termasuk pada pasien yang masuk dan keluar pada 24 jam sebelumnya. Umumnya pelaporan dimulai dari pukul 00.01 pagi dan berakhir di pukul 24.00 malam. Dalam sensus ini, bayi dihitung terpisah dari pasien dewasa dan anak-anak [4]. Data sensus harian digunakan sebagai sumber data dalam pembuatan indikator pelayanan rawat inap yang selanjutnya digunakan dalam perhitungan BOR, AvLOS, BTO dan TOI.

c. Analisa Sistem

Analisis sistem merupakan tahapan penting dalam pengembangan sistem informasi yang bertujuan untuk memahami permasalahan yang ada pada sistem lama serta merancang solusi sistem baru yang lebih efektif dan efisien [5].

d. Metode *Waterfall*

Metode air terjun atau yang dikenal sebagai *waterfall* sering disebut sebagai *classic life cycle* atau Linear Sequential Model. Model ini menggambarkan pendekatan pengembangan perangkat lunak yang bersifat sistematis dan berurutan. Proses pengembangan dimulai dari tahap spesifikasi kebutuhan

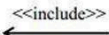
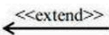
pengguna, kemudian dilanjutkan dengan tahapan perencanaan (*planning*), permodelan (*modelling*), konstruksi (*construction*), hingga penyerahan sistem kepada pengguna (*deployment*), yang selanjutnya diakhiri dengan tahap dukungan dan pemeliharaan (*maintenance*) terhadap perangkat lunak yang telah dihasilkan [6].

e. *Use Case Diagram*

Use Case Diagram merupakan salah satu komponen utama dalam pemodelan sistem berbasis *Unified Modeling Language* (UML). Diagram ini digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem, berdasarkan fungsi-fungsi atau layanan utama yang disediakan oleh sistem tersebut. Setiap interaksi digambarkan dalam bentuk *use case* (kasus penggunaan), yang merepresentasikan satu fitur atau proses penting dalam sistem [3].

Aktor dalam *Use Case Diagram* bisa berupa pengguna sistem, perangkat lain, atau entitas luar yang berinteraksi langsung dengan sistem. Sementara itu, *use case* menunjukkan aktivitas atau layanan yang bisa dilakukan oleh aktor melalui sistem. *Use Case Diagram* adalah sebagai berikut:

Gambar 1. Simbol *Use Case Diagram*

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

f. Activity Diagram

Activity diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan alur kerja (*workflow*) atau urutan aktivitas dalam suatu proses bisnis atau sistem. Diagram ini menekankan proses aliran logika dari satu aktivitas ke aktivitas lain, termasuk percabangan, pengulangan, dan sinkronisasi proses [5]. *Activity* Diagram adalah sebagai berikut:

Gambar 2. Simbol *Activity* Diagram

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

g. Class Diagram

Class diagram adalah salah satu pemodelan yang cukup penting dalam UML, fungsinya adalah untuk membuat sebuah *logical models* dari sebuah sistem. Sebuah *class* diagram akan menunjukkan bagaimana skema dari arsitektur sebuah sistem yang sedang dirancang. *Class* diagram digambarkan dengan *class* yang berisi atribut dan *method*, setiap *class* akan dihubungkan dengan sebuah garis disebut Asosiasi [7].

h. MySQL

MySQL adalah sebuah *Relational Database Management System* (RDBMS) atau sistem manajemen basis data relasional yang paling populer dan banyak digunakan di seluruh dunia. MySQL menggunakan bahasa standar *Structured Query Language* (SQL) untuk mengelola, memanipulasi, dan mengakses data di dalam tabel-tabel yang saling terhubung. MySQL dikembangkan oleh MySQL AB, sebuah perusahaan Swedia, yang kemudian diakuisisi oleh Sun

Microsystems, dan akhirnya

menjadi bagian dari Oracle Corporation [5].

i. Pengujian *BlackBox*

Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada fungsi sistem dari sisi pengguna, tanpa mengetahui struktur internal, kode sumber, atau alur logika dari program tersebut. Pengujian ini dilakukan untuk memeriksa apakah input tertentu menghasilkan output yang sesuai, berdasarkan spesifikasi dan kebutuhan sistem.

Tujuan utama dari pengujian *black box* adalah untuk memastikan bahwa seluruh fungsi sistem berjalan sesuai dengan spesifikasi yang telah ditetapkan. Selain itu, pengujian ini bertujuan untuk menemukan berbagai kesalahan pada fungsi sistem, seperti kesalahan pada proses input dan output, kesalahan antarmuka, kesalahan akses basis data, serta kesalahan fungsional lainnya. Pengujian *black box* juga dilakukan untuk menjamin bahwa setiap fitur sistem yang diuji mampu menghasilkan keluaran yang benar sesuai dengan kebutuhan pengguna [5].

METODE PENELITIAN

Penelitian ini menerapkan metode *Waterfall*, yaitu salah satu model pengembangan perangkat lunak yang bersifat klasik, sistematis, dan terstruktur. Metode ini dilaksanakan secara berurutan melalui beberapa tahapan, mulai dari analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan. Pendekatan *Waterfall* dipilih karena sesuai untuk pengembangan aplikasi dengan alur kerja yang jelas serta kebutuhan sistem yang dapat dirumuskan secara lengkap sejak tahap awal pengembangan.

Berikut tahapan-tahapan dalam metode *Waterfall* yang diterapkan dalam penelitian ini:

1. Analisis Kebutuhan

Pada tahap ini dilakukan proses pengumpulan data dan identifikasi kebutuhan yang akan digunakan sebagai dasar pada tahap selanjutnya. Teknik pengumpulan data yang diterapkan meliputi wawancara dengan petugas dan observasi langsung di lapangan.

2. Desain Sistem

Pada tahap ini perancangan sistem menggunakan desain data base, antarmuka pengguna, serta diagram alur sistem seperti, Data Flow Diagram (DFD), *use case* diagram, *activity* diagram, dan *class* diagram.

3. Implementasi

Setelah tahap desain selesai, dilakukan proses pengkodean dengan mengimplementasikan desain sistem ke dalam bentuk aplikasi yang dapat berjalan sesuai kebutuhan. Pengembangan dilakukan menggunakan bahasa pemrograman PHP dengan framework CodeIgniter 4 serta basis data MySQL. Proses produksi dan pengujian awal aplikasi dijalankan pada server Apache secara lokal melalui aplikasi XAMPP.

4. Pengujian

Pengujian sistem dilakukan dengan menggunakan metode *Black Box Testing*, yaitu teknik pengujian yang berfokus pada aspek fungsional sistem tanpa memperhatikan struktur kode program. Pengujian ini bertujuan untuk memastikan setiap fitur berfungsi sesuai dengan yang diharapkan.

5. Pemeliharaan

Pemeliharaan perangkat lunak merupakan tahap lanjutan setelah sistem diuji, digunakan, dan diserahkan kepada pengguna, yang

mencakup proses evaluasi untuk mengidentifikasi adanya bug maupun kebutuhan tambahan dari pengguna. Tahap ini

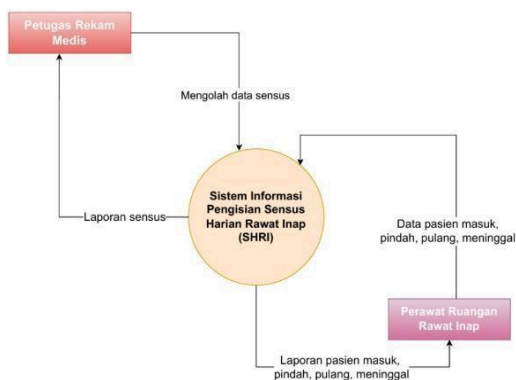
meliputi kegiatan perbaikan kesalahan pada kode program, penyempurnaan desain sistem, serta penyesuaian terhadap spesifikasi atau kebutuhan baru, sehingga sistem tetap berjalan stabil dan mampu beradaptasi dengan perkembangan kebutuhan pengguna.

HASIL DAN PEMBAHASAN

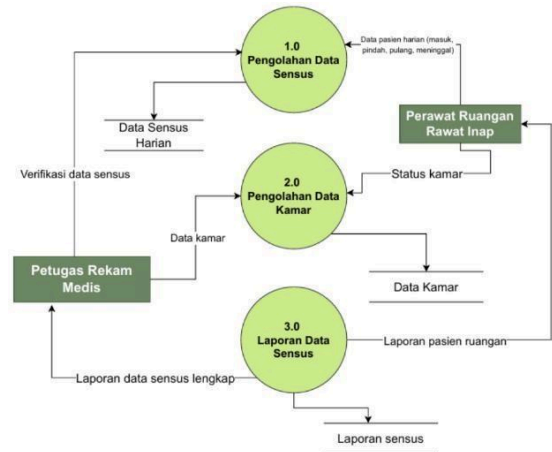
a. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) yaitu diagram yang menggunakan notasi berupa simbol-simbol untuk menggambarkan arus data pada suatu sistem. DFD digambarkan mulai dari diagram konteks, level 0, level 1 hingga level 2.

Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke dalam sistem atau output dari sistem yang memberi gambaran tentang keseluruhan sistem. DFD level 0 menjelaskan satu lingkaran besar yang mewakili sistem yang berinteraksi dengan eksternal entitas. Semua entitas yang ada pada DFD termasuk juga aliran datanya diarahkan langsung kepada sistem. DFD level 1 adalah semua proses dipecah menjadi sub-proses kemudian dirinci lebih lengkap dan detail. DFD Level 2 merupakan gambaran dengan penjelasan yang lebih lebih detail dari DFD level 1 [8].



Gambar 3. Diagram Konteks



Gambar 4. DFD Level 0

Gambar 5. DFD Level 1

b. Use Case Diagram

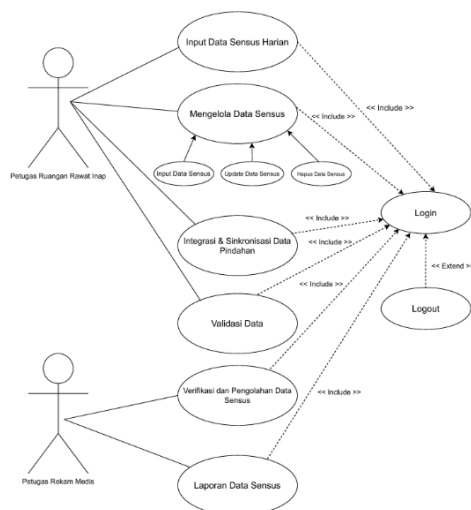
Use case diagram menggambarkan fungsional yang diharapkan dari sebuah sistem. *Use case* bertujuan untuk mempresentasikan interaksi antara actor dengan sistem. Actor adalah suatu entitas manusia yang berinteraksi dengan sistem untuk melakukan pekerjaanpekerjaan tertentu [9].

Use Case Diagram ini menggambarkan dua aktor utama, yaitu Petugas Ruang Rawat Inap dan Petugas Rekam Medis, yang masing- masing memiliki peran dan hak akses berbeda dalam sistem. Petugas Ruang Rawat Inap memiliki akses untuk melakukan *login/logout*. Actor ini dapat

melakukan input data sensus

harian, serta mengelola data sensus yang mencakup kegiatan menambah, memperbarui, dan menghapus data sensus. Selain itu, petugas juga berperan dalam integrasi dan sinkronisasi data pindahan serta melakukan validasi data guna memastikan data sensus yang diinput sesuai dan akurat.

Sementara itu, Petugas Rekam Medis memiliki hak akses untuk melakukan *login/logout*. Aktor ini bertanggung jawab dalam verifikasi serta pengolahan data sensus yang telah diinput oleh petugas ruangan. Selain itu, petugas rekam medis juga dapat melihat dan menghasilkan laporan data sensus sebagai bahan evaluasi dan dokumentasi.



Gambar 6. Use Case Diagram

c. Activity Diagram

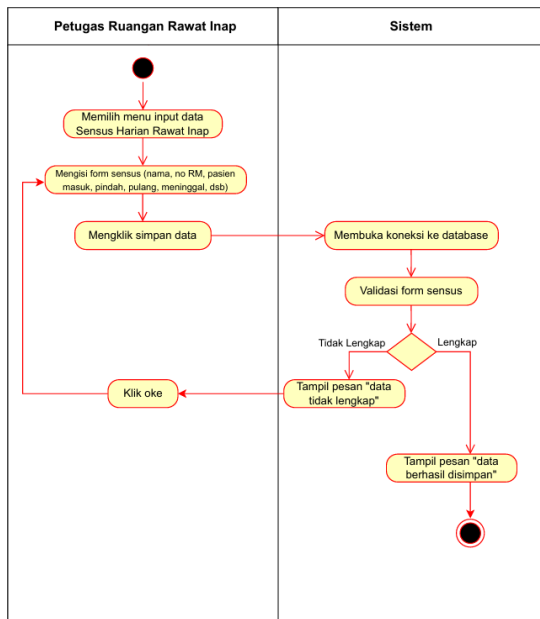
Activity diagram adalah pemodelan yang dilakukan pada suatu sistem dan menggambarkan aktivitas sistem berjalan. *Activity* diagram digunakan sebagai penjelasan aktivitas program tanpa melihat koding atau tampilan [10].

Activity diagram pada sistem informasi Sensus Harian Rawat Inap ini dapat diuraikan dalam *activity* diagram dengan menjelaskan *workflow* dari

sistem informasi Sensus Harian Rawat Inap yang diimplementasikan dalam aplikasi yang dibuat. Hal ini bertujuan untuk memahami alur kerja dari masing-masing komponen dalam sistem informasi yang dibuat. Jadi gambaran umum *activity* diagram sistem informasi Sensus Harian Rawat Inap dapat dilihat sebagai berikut :

1. Melihat urutan proses dari sistem informasi Sensus Harian Rawat Inap.
2. Memahami alur proses data keseluruhan dari sistem informasi Sensus Harian Rawat Inap.
3. Menggambarkan suatu aktifitas dari sistem informasi Sensus Harian Rawat Inap
4. Menguraikan aktifitas dari sistem informasi Sensus Harian Rawat Inap.

Gambar 7. Activity Diagram Halaman Login



Gambar 8. Activity Diagram Input Data

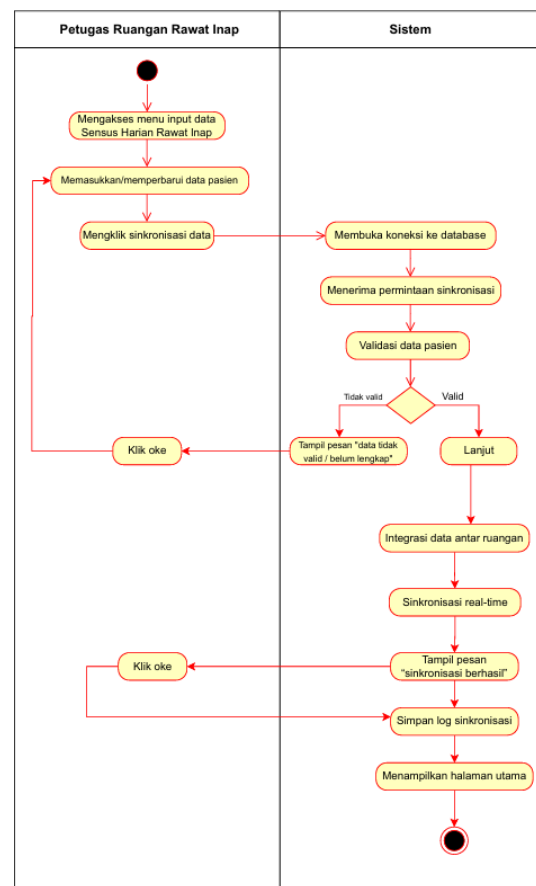
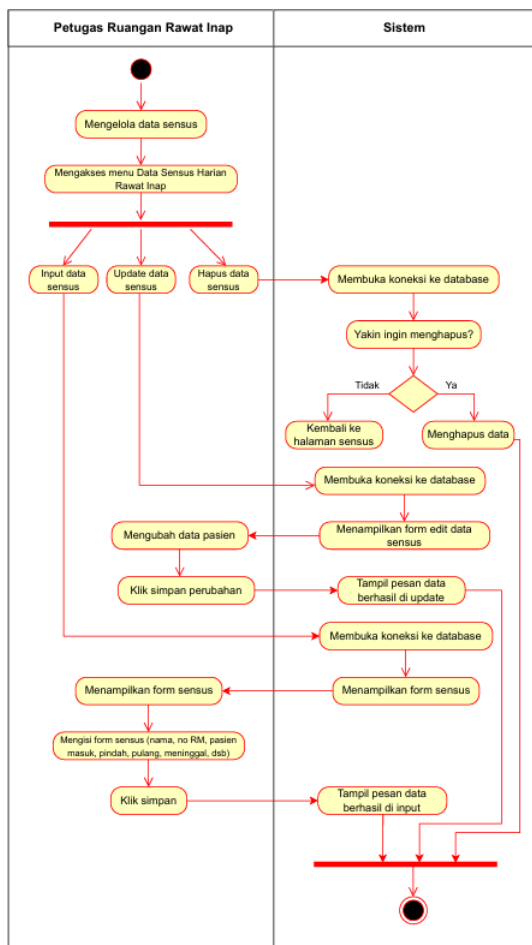
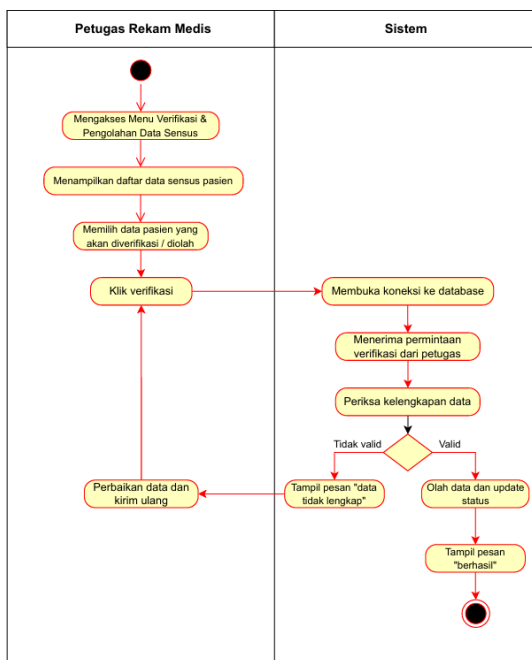


Diagram Integrasi & Sinkronisasi Data Pindahan

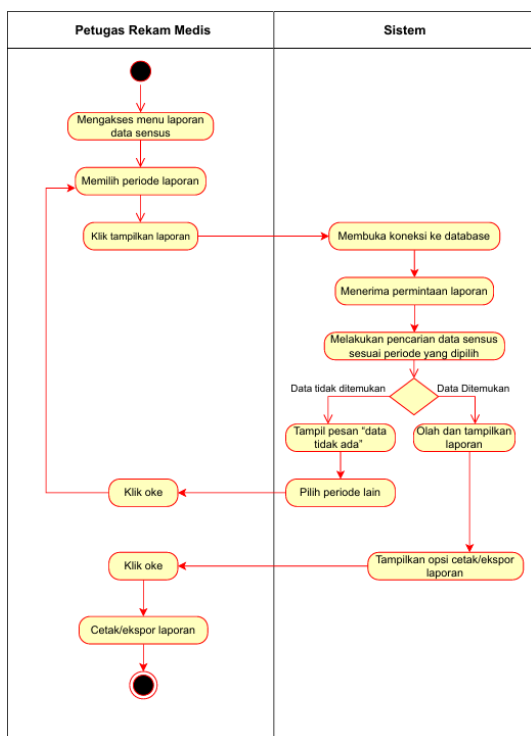


Gambar 9. Activity Diagram Mengelola Data

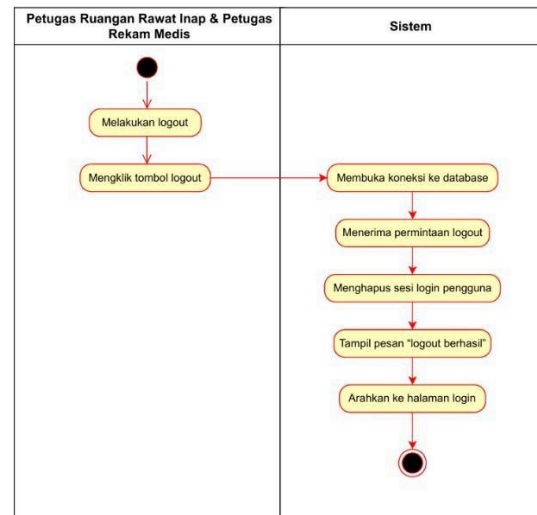
Gambar 11. Activity Diagram Validasi Data



Gambar 12. *Activity Diagram* Verifikasi dan Pengolahan Data



Gambar 13. *Activity Diagram* Laporan Data



Gambar 14. *Activity Diagram* Logout

d. *Class Diagram*

Class diagram menunjukkan bagaimana skema dari arsitektur sebuah sistem yang sedang dirancang [7]. *Class Diagram* ini menggambarkan struktur kelas dan hubungan antar entitas pada sistem Sensus Harian Rawat Inap. Terdiri dari beberapa kelas utama, yaitu Pegawai, Petugas Ruangan Rawat Inap, Petugas Rekam Medis, Ruangan, Pasien, Sensus Harian Rawat Inap, dan Audit Trail.

Kelas Pegawai berfungsi sebagai kelas induk yang menyimpan atribut umum seperti *Id_Pegawai*, *Nama_Pegawai*, *Jabatan*, *Username*, dan *Password*, serta memiliki fungsi login, logout, dan menampilkan profil. Kelas ini diturunkan ke dua kelas turunan, yaitu Petugas Ruangan Rawat Inap dan Petugas Rekam Medis, yang masing-masing memiliki atribut dan peran khusus sesuai tugasnya.

Petugas Ruangan Rawat Inap bertanggung jawab dalam mencatat data sensus harian rawat inap, termasuk data pasien dan data ruangan. Petugas ini memiliki relasi dengan kelas Sensus Harian Rawat Inap, di mana satu petugas dapat

mencatat banyak data sensus. Selain itu, setiap aktivitas yang dilakukan oleh petugas akan dicatat ke

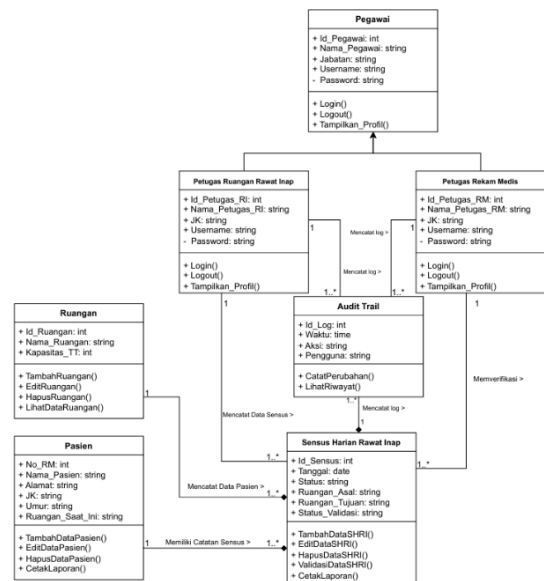
dalam kelas Audit Trail sebagai log aktivitas sistem.

Petugas Rekam Medis berperan dalam melakukan verifikasi dan pengolahan data sensus harian rawat inap yang telah diinput. Satu petugas dapat mencatat banyak data sensus serta terhubung dengan Audit Trail untuk mencatat setiap proses verifikasi dan perubahan data.

Kelas Ruangan menyimpan informasi terkait ruang rawat inap seperti *Id_Ruangan*, *Nama_Ruangan*, dan *Kapasitas_TT*. Kelas ini berelasi dengan Sensus Harian Rawat Inap, di mana satu ruangan dapat memiliki banyak catatan sensus.

Kelas Pasien menyimpan data pasien, meliputi *Nomor_RM*, *Nama_Pasien*, *Alamat*, *JK*, *Umur*, dan *Ruangan_Saat_Ini*. Setiap pasien dapat memiliki lebih dari satu catatan sensus harian, sehingga membentuk relasi dengan kelas Sensus Harian Rawat Inap.

Sementara itu, kelas Audit Trail berfungsi untuk mencatat seluruh aktivitas dan perubahan data yang dilakukan oleh pengguna sistem, seperti waktu, aksi, dan pengguna yang melakukan perubahan. Relasi ini memastikan transparansi, keamanan, dan kemudahan dalam pelacakan riwayat aktivitas sistem.



Gambar 15. Class Diagram

e. Halaman Login

Halaman *Login* merupakan bagian dari sistem yang berfungsi sebagai mekanisme autentikasi pengguna melalui proses pemasukan *username* dan *password*. Melalui mekanisme ini, sistem dapat mengenali identitas pengguna serta menetapkan hak akses sesuai dengan peran dan kewenangan yang dimiliki.

Halaman login bertujuan untuk menjamin keamanan sistem dengan membatasi akses hanya kepada pengguna yang memiliki otorisasi. Pada sistem ini, halaman *login* digunakan oleh perawat ruangan dan petugas rekam medis, di mana masing-masing pengguna diberikan hak akses sesuai dengan tugas dan tanggung jawabnya.

Gambar 16. Tampilan Halaman Login

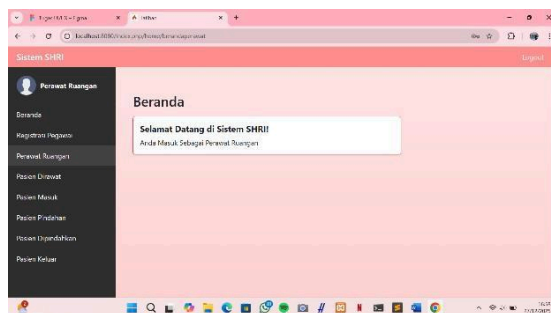
f. Halaman Beranda

Halaman Beranda pada Sistem SHRI merupakan halaman utama yang ditampilkan setelah pengguna berhasil melakukan *login*. Halaman beranda pada sistem ini dibagi menjadi dua jenis berdasarkan peran pengguna, yaitu Halaman Beranda Perawat Ruangan dan Halaman Beranda Petugas RM.

Pada Halaman Beranda Perawat Ruangan, sistem menampilkan keterangan bahwa pengguna masuk sebagai perawat ruangan. Halaman ini dilengkapi dengan menu navigasi yang memudahkan perawat ruangan dalam mengakses fitur pengelolaan data pasien, seperti pasien dirawat, pasien masuk, pasien pindahan, pasien dipindahkan, dan pasien keluar.

Sementara itu, pada Halaman Beranda Petugas RM, sistem menampilkan keterangan bahwa pengguna masuk sebagai petugas RM. Menu navigasi pada halaman ini digunakan untuk mengakses fitur yang berkaitan dengan rekapitulasi laporan SHRI.

Beranda Petugas RM



Gambar 17. Tampilan Halaman Beranda Perawat Ruangan



Gambar 18. Tampilan Halaman

g. Halaman Registrasi Pegawai

Halaman Registrasi Pegawai digunakan untuk mengelola data pegawai. Setiap data pegawai yang ditampilkan dapat dikelola melalui menu Aksi, yang menyediakan fitur Edit dan Hapus untuk memperbarui atau menghapus data pegawai yang telah tersimpan di dalam sistem.

Selain itu, halaman ini dilengkapi dengan tombol Tambah yang berfungsi untuk menambahkan data pegawai baru ke dalam sistem. Melalui halaman registrasi pegawai ini, proses pengelolaan data pegawai dapat dilakukan secara terstruktur dan sistematis, sehingga mendukung ketertiban administrasi serta memudahkan pengaturan hak akses pengguna sesuai dengan jabatan masing-masing.

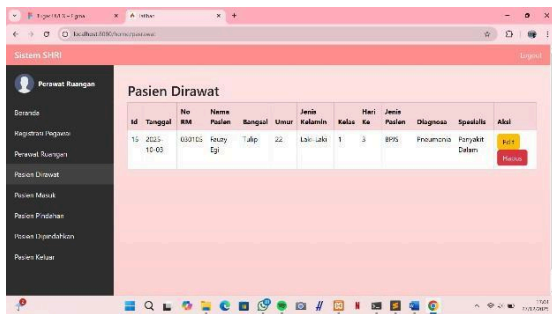


Gambar 19. Tampilan Halaman Registrasi Pegawai

h. Halaman Pasien Dirawat

Halaman ini digunakan untuk menampilkan dan mengelola data pasien yang sedang menjalani rawat inap. Halaman ini memuat informasi penting pasien, seperti identitas pasien, tanggal perawatan, diagnosis, kelas perawatan, serta jenis penjamin. Fitur Edit digunakan untuk memperbarui data pasien apabila terjadi perubahan atau koreksi informasi, sehingga data

yang tersimpan tetap akurat dan sesuai dengan kondisi terkini.

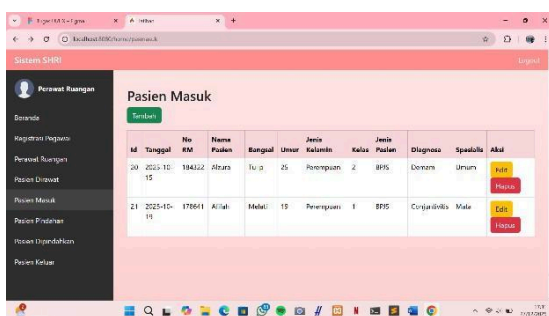


Gambar 20. Tampilan Halaman Pasien Dirawat

i. Halaman Pasien Masuk

Halaman ini digunakan untuk mencatat dan mengelola data pasien yang baru masuk atau mulai menjalani perawatan. Halaman ini dilengkapi dengan beberapa fitur utama, yaitu Tambah, Edit, dan Hapus. Fitur Tambah digunakan untuk memasukkan data pasien baru ke dalam sistem saat pasien pertama kali masuk.

Fitur Edit berfungsi untuk memperbarui atau memperbaiki data pasien apabila terjadi perubahan atau kesalahan pencatatan. Sementara itu, fitur Hapus digunakan untuk menghapus data pasien yang tidak valid atau tercatat ganda.



Gambar 21. Tampilan Halaman Pasien Masuk

j. Halaman Pasien Pindahan

Halaman ini digunakan untuk menangani mutasi pasien antar ruang perawatan atau bangsal secara efisien. Pada halaman ini sudah terintegrasi penuh dengan fitur Pasien Masuk, yang

memungkinkan terjadinya sinkronisasi data secara otomatis. Ketika pasien memerlukan perpindahan unit, petugas hanya diwajibkan mengisi bagian "Pindahan Dari" untuk menentukan asal bangsal pasien pada aksi edit.

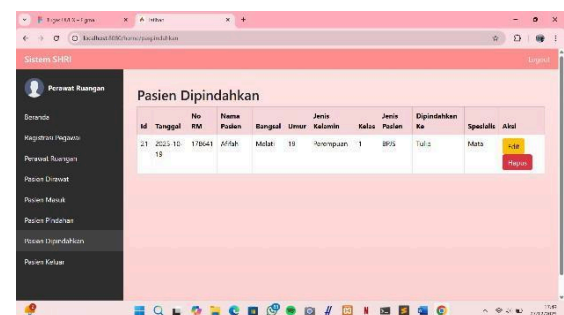


Gambar 22. Tampilan Halaman Pasien Pindahan

k. Halaman Pasien Dipindahkan

Halaman ini digunakan untuk mengelola data pasien yang keluar dari suatu unit menuju unit perawatan lain. Pada halaman ini sudah terintegrasi penuh dengan fitur Pasien Masuk, yang memungkinkan terjadinya sinkronisasi data secara otomatis.

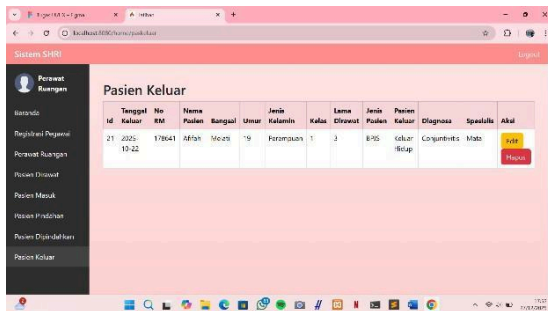
Petugas hanya diwajibkan mengisi kolom "Dipindahkan Ke" pada aksi edit untuk menentukan tujuan bangsal baru pasien, sementara informasi lainnya akan terisi secara otomatis oleh sistem berdasarkan riwayat pendaftaran awal.



Gambar 23. Tampilan Halaman Pasien Dipindahkan

l. Halaman Pasien Keluar

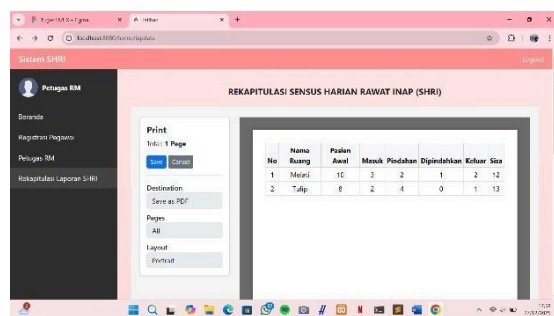
Halaman ini mengintegrasikan halaman Pasien Dipindahkan dan halaman Pasien Keluar sebagai instrumen kendali dalam memantau alur mutasi perawatan pasien secara sistematis. Implementasi halaman-halaman ini bertujuan untuk memastikan validitas data rekam medis elektronik dan efisiensi operasional tenaga medis di unit perawatan.



Gambar 24. Tampilan Halaman Pasien Keluar

m. Halaman Cetak Laporan

Halaman ini hanya dapat diakses oleh pengguna dengan peran Petugas RM setelah melakukan login ke dalam sistem. Fitur ini digunakan untuk menampilkan dan mencetak laporan rekapitulasi SHRI sebagai bahan dokumentasi, pembuatan grafik *Barber Johnson*, dan pelaporan data pelayanan rumah sakit.



Gambar 25. Tampilan Halaman Cetak Laporan

n. Uji BlackBox

Tabel *BlackBox* ini menjelaskan modul yang di uji melalui *test case* yang dilakukan dan hasil pengujiannya. Hasil menunjukkan bahwa sistem berhasil

menjalankan fungsinya tanpa ditemukan kesalahan.

No	Modul Yang Di Uji	Test Case	Hasil Pengujian
1	Halaman Login	Mengisi Username dan Password	Berhasil
2	Halaman Registrasi Perawat	Tambah, Edit, Hapus	Berhasil
3	Halaman Pasien Dirawat	Edit	Berhasil
4	Halaman Pasien Masuk	Tambah, Edit, Hapus	Berhasil
5	Halaman Pasien Pindahan	Edit	Berhasil
6	Halaman Pasien Dipindahkan	Edit	Berhasil
7	Halaman Pasien Keluar	Edit	Berhasil

Tabel 1. Uji *BlackBox*

SIMPULAN

Implementasi desain antarmuka sistem Sensus Harian Rawat Inap (SHRI) melalui pendekatan metode *Waterfall* ini mampu menyederhanakan alur pencatatan mutasi pasien termasuk data pasien masuk, pindahan, hingga pasien keluar yang kini saling terintegrasi untuk meminimalkan risiko ketidaklengkapan data manual.

Pengembangan aplikasi yang memanfaatkan bahasa pemrograman PHP dengan *framework* CodeIgniter 4 serta basis data MySQL ini menjamin kemudahan akses bagi pengguna, baik perawat di ruangan maupun petugas

rekam medis, dalam menjalankan tugas administratif secara lebih terstruktur. Hasil pengujian dengan metode *Black Box*

Testing menunjukkan bahwa seluruh fungsi utama sistem, mulai dari halaman login hingga pengelolaan data pasien, berjalan dengan baik tanpa ditemukan kesalahan fungsional. Secara keseluruhan, aplikasi ini meningkatkan validitas data rekam medis, meminimalkan kendala ketidaklengkapan pengisian manual, dan membantu rumah sakit dalam menghasilkan laporan rekapitulasi SHRI yang akurat untuk mendukung evaluasi pelayanan.

UCAPAN TERIMA KASIH

1. Puji syukur Kepada Tuhan Yang Maha Esa

Karena atas segala rahmat dan karunia-Nya, sehingga penyusunan jurnal ini dapat diselesaikan dengan baik.

2. Institusi Universitas Awal Bros
Dengan penyediaan fasilitas pendidikan yang memadai dan lingkungan akademik yang mendukung proses pembelajaran mahasiswa.

3. Ibu Desfa Anisa dan Bapak Miftahul Ilmi

Selaku dosen pengampu mata kuliah Analisis dan Perancangan Sistem Informasi yang telah meluangkan waktu, pikiran, dan perhatian untuk memberikan pengarahan dan bimbingan sehingga jurnal ini dapat terselesaikan dengan baik.

DAFTAR PUSTAKA

- [1] Kemenkes, “Peraturan Menteri Kesehatan Republik Indonesia Nomor 3 Tahun 2020 tentang Klasifikasi dan Perizinan Rumah Sakit,” *Peraturan Menteri Kesehatan Republik Indonesia*, no. 3, pp. 1–80, 2020.
- [2] T. Diniyah, R. D. Pratiwi, D. Iii, R. Medis, S. Vokasi, and U. Gadjah, “Desain Antarmuka Sistem Informasi Sensus Harian Rawat Inap di Rs Krakatau Medika Cilegon,” vol. 5, no. 1, 2020.
- [3] Emilia and M. Ilmi, “PERANCANGAN SISTEM INFORMASI INVENTARISASI BARANG GUDANG BERBASIS DESKTOP,” vol. 2, pp. 10–20, 2025.
- [4] M Luthfi Fauzi, Yuda Syahidin, and Syaikhul Wahab, “Perancangan Sistem Informasi Grafik Barber Johnson Dalam Mengukur Efisiensi Rumah Sakit Menggunakan Microsoft Visual Studio 2013,” *INFOKES (Informasi Kesehatan)*, vol. 5, no. 2, pp. 29–37, 2021, doi: 10.56689/infokes.v5i2.462.
- [5] D. Ardian and M. Ilmi, “ANALISA DAN PERANCANGAN APLIKASI MANAJEMEN LOGISTIK KOPI,” vol. 2, pp. 1–9, 2025.
- [6] A. A. Wahid, “Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi,” pp. 1–5, 2020.
- [7] W. Aliman, “PERANCANGAN PERANGKAT LUNAK UNTUK MENGGAMBAR DIAGRAM BERBASIS ANDROID,” vol. 6, no. 6, 2021.
- [8] R. A. Y. Manurung and A. D. Manuputty, “PERANCANGAN SISTEM INFORMASI LEMBAGA KEMAHASISWAAN UNIVERSITAS KRISTEN SATYA WACANA SALATIGA,” 2020.
- [9] M. Rahmatuloh and M. R. Revanda, “RANCANG BANGUN SISTEM INFORMASI JASA PENGIRIMAN BARANG PADA PT. HALUAN INDAH TRANSPORINDO BERBASIS WEB,” vol. 14, no. 1, pp. 54–59, 2022.
- [10] V. Puturuhi, “SISTEM INFORMASI MANAJEMEN PENELITIAN DAN PENGABDIAN PNPB PADA POLITEKNIK NEGERI AMBON,” vol. 12, no. 1, pp. 553–560, 2022.