

LAPORAN REKAYASA PERANGKAT LUNAK
OPTIMISASI SISTEM REGISTRASI RUMAH SAKIT MELALUI
REKAYASA PERANGKAT LUNAK

DOSEN PENGAMPU:

Bhustomy Hakim, S.S.I., M.Eng.



Anggota :

Reynaldi	32230009
Varel Gabriel Wungow	32230062
Alexander Leon	32230082
Davin Cleon	32230082

Universitas Bunda Mulia
Fakultas Teknologi dan Desain
Informatika
Kampus Serpong
2024

DAFTAR ISI

BAB I	3
PENDAHULUAN	3
1.1 Latar Belakang	3
1.2 Rumusan Masalah	3
1.3 Ruang Lingkup / Batasan Masalah	4
1.4 Tujuan dan Manfaat	4
• Tujuan :	4
• Manfaat:	4
BAB II LANDASAN TEORI.....	6
2.1 Teori Umum.....	6
• Rekayasa Perangkat Lunak	6
• Efisiensi operasional dan kepuasan pasien	6
• Registrasi	6
• Terintegrasi.....	6
• Operasional	7
2.2 Teori Khusus	7
• Systems Development Life Cycle (SDLC).....	7
• Waterfall Method	7
• Data Flow Diagram (DFD)	8
• Unified Modelling Language (UML)	9
• Entity Relationship Diagram (ERD)	9
BAB III ANALISIS DAN PERANCANGAN.....	10
3.1 Desain Proses Terstruktur	10
• Diagram Level 0	10
• Diagram Level 1	11
• Diagram Level 2	11
3.2 Desain OOP(UML).....	12
• Use case.....	12
• Activity Diagram	13
• Activity Diagram Pembelian Obat.....	14
3.3 Desain ERD	15
• Entitas dan Atribut Utama:	15
• Relasi Antar Entitas:.....	16
DAFTAR PUSTAKA	18

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem registrasi di rumah sakit merupakan salah satu kunci dalam memberikan pelayanan kesehatan yang efisien dan berkualitas. Proses ini mencakup pendaftaran pasien, penjadwalan, hingga pengelolaan informasi medis yang sangat menentukan alur operasional rumah sakit. Namun, banyak rumah sakit masih menghadapi berbagai permasalahan, seperti antrian panjang, kesalahan input data, serta ketidaknyamanan bagi pasien dan staf. Hal ini biasanya disebabkan oleh penggunaan sistem manual yang kurang optimal, yang tidak mampu mengimbangi kuantitas dan kerumitan data pasien.

Dengan perkembangan teknologi informasi, rekayasa perangkat lunak menawarkan solusi yang dapat mengoptimalkan sistem registrasi rumah sakit. Pengembangan sistem yang lebih terintegrasi, otomatis, dan user-friendly dapat meningkatkan efisiensi proses registrasi, meminimalisir kesalahan. Laporan ini akan membahas permasalahan dalam sistem registrasi rumah sakit pada umumnya dan bagaimana penerapan rekayasa perangkat lunak dapat menjadi solusi untuk mengoptimalkannya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang kita buat, berikut adalah rumusan masalah yang kita temukan :

- Permasalahan utama yang dihadapi dalam proses registrasi rumah sakit
- Bagaimana pengimplementasian rekayasa perangkat lunak dalam mengatasi permasalahan dalam registrasi rumah sakit
- Keefektifan penerapan rekayasa perangkat lunak dalam meningkatkan optimalitas dan efisiensi sistem registrasi dan operasional rumah sakit

1.3 Ruang Lingkup / Batasan Masalah

Agar pembahasan pada laporan ini lebih terfokus pada pokok masalah yang kita bahas, berikut adalah beberapa batasan masalah yang kami bahas :

- Analisis kita berfokus pada proses registrasi pasien di rumah sakit, yang mencakup pendaftaran, penjadwalan, dan pengelolaan informasi pasien.
- Pengimplementasian rekayasa perangkat lunak yang kita bahas hanya mencakup sistem registrasi yang terintegrasi.
- Pembahasan kita dibatasi pada rumah sakit dengan skala menengah hingga besar, dengan asumsi kuantitas dan kompleksitas pasien yang tinggi, sehingga dibutuhkan sistem yang memadai.
- Implementasi yang kita bahas lebih berfokus pada rumah sakit di wilayah dengan infrastruktur teknologi yang memadai, seperti di perkotaan.

1.4 Tujuan dan Manfaat

- **Tujuan :**
 - **Menganalisis Masalah Registrasi:** Mengidentifikasi dan menganalisis permasalahan yang ada dalam sistem registrasi rumah sakit yang dapat memengaruhi kualitas pelayanan.
 - **Mengusulkan Solusi Teknologi:** Mengembangkan rekomendasi solusi berbasis rekayasa perangkat lunak untuk meningkatkan efisiensi dan efektivitas sistem registrasi.
 - **Meningkatkan Kualitas Pelayanan:** Menyusun strategi yang dapat meningkatkan kualitas pengalaman pasien dalam melakukan registrasi di rumah sakit.
 - **Menilai Dampak Implementasi:** Mengukur dan mengevaluasi dampak penerapan sistem baru terhadap operasional rumah sakit dan kepuasan pasien.
- **Manfaat:**

- **Peningkatan Efisiensi:** Sistem registrasi yang lebih baik akan mengurangi waktu tunggu pasien dan mempercepat proses pendaftaran.
- **Pengurangan Kesalahan Data:** Dengan sistem otomatis, kemungkinan kesalahan input dan duplikasi informasi dapat diminimalkan, meningkatkan akurasi data.
- **Kepuasan Pasien yang Lebih Tinggi:** Pengalaman registrasi yang lebih baik berpotensi meningkatkan kepuasan pasien dan loyalitas terhadap rumah sakit.
- **Optimisasi Sumber Daya Staf:** Staf dapat lebih fokus pada tugas pelayanan kesehatan utama, bukan pada pekerjaan administratif yang memakan waktu.
- **Data yang Lebih Terintegrasi:** Pengumpulan dan pengelolaan data yang lebih efisien akan memberikan informasi yang lebih akurat untuk pengambilan keputusan manajerial di rumah sakit.

BAB II

LANDASAN TEORI

2.1 Teori Umum

- **Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak atau software engineering merupakan Ilmu yang berfokus pada metode, teknik, dan prinsip-prinsip dalam pengembangan perangkat lunak yang efisien.

Dalam konteks optimisasi registrasi rumah sakit, RPL menciptakan sistem yang lebih optimal, efektivitas, dan user-friendly dalam proses registrasi serta pengelolaan data pasien. Berbagai metodologi dalam pengembangan perangkat lunak, seperti Agile dan Waterfall bisa diterapkan untuk memastikan sistem dapat sesuai dengan kebutuhan dalam penyelesaian masalah kami.(MOHAMAD SIDIQ,2020)

- **Efisiensi operasional dan kepuasan pasien**

Efisiensi operasional rumah sakit mengoptimalkan sumber daya untuk mempercepat proses registrasi, mengurangi waktu tunggu, dan meningkatkan kenyamanan pasien. Pengalaman positif saat registrasi awal berdampak besar pada kepuasan pasien secara keseluruhan.(BITHEALTH,2024)

- **Registrasi**

Registrasi adalah proses pendaftaran atau pencatatan seseorang, produk, atau organisasi ke dalam sebuah sistem resmi untuk tujuan pengakuan atau pengesahan. Dalam konteks pendidikan, registrasi sering merujuk pada pendaftaran siswa untuk mengikuti program tertentu. Dalam layanan digital, registrasi berarti pembuatan akun pengguna baru. Proses ini biasanya mencakup pengumpulan informasi penting seperti identitas, kontak, atau preferensi pengguna.(SPADA,2022)

- **Terintegrasi**

Terintegrasi berarti menyatukan berbagai elemen atau sistem yang sebelumnya terpisah agar dapat bekerja bersama dalam satu kesatuan yang harmonis. Sistem terintegrasi memungkinkan kolaborasi yang lebih efisien, misalnya antara departemen yang berbeda dalam sebuah organisasi. Dengan integrasi, setiap bagian dapat berbagi data dan berfungsi secara terpadu,

sehingga meningkatkan efisiensi dan keakuratan operasional.(DIBIMBING,2023)

- **Operasional**

"Operasional" mengacu pada proses atau kegiatan yang terkait dengan pelaksanaan tugas-tugas rutin dalam suatu perusahaan atau organisasi. Ini mencakup berbagai aktivitas yang diperlukan untuk menjaga kelancaran, pengendalian, dan efisiensi operasi bisnis sehari-hari, seperti produksi, distribusi, dan administrasi.(BIDANG USAHA,2024)

2.2 Teori Khusus

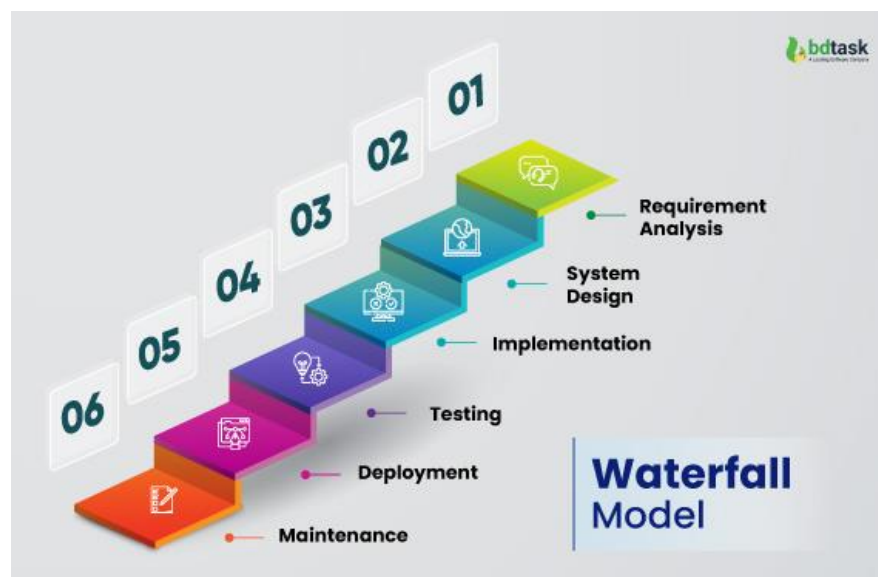
- **Systems Development Life Cycle (SDLC)**

Siklus hidup pengembangan perangkat lunak (SDLC) adalah proses yang hemat biaya dan hemat waktu yang dipakai oleh tim pengembangan untuk mendesain serta membangun perangkat lunak yang berkualitas tinggi. **Tujuan SDLC** adalah meminimalkan risiko proyek melalui perencanaan ke depan, agar perangkat lunak memenuhi harapan pelanggan selama produksi dan seterusnya.

Metode ini menjelaskan serangkaian langkah-langkah yang membagi proses pengembangan perangkat lunak menjadi tugas yang dapat ditetapkan, selesaikan, dan diukur.

Source : <https://aws.amazon.com/id/what-is/sdlc/>

- **Waterfall Method**



SDLC metode waterfall adalah sebuah project design method yang sederhana, dimana konsepnya adalah step-by-step proses dimana step berikutnya tidak bisa jalan sebelum proses “sekarang” belum selesai. Tahap-tahap proses itu termasuk:

- 1) **Melihat costumer requirement**, ini menjadi tahap pertama yang cocok karena dengan mengetahui terlebih dahulu apa yang costumer butuhkan, project akan memiliki tujuan yang jelas dan dengan begitu, tugas yang akan dilaksanakan sudah menjadi jelas.
- 2) **Analisis sumberdaya**; dimana di tahap ini akan memastikan project akan berjalan lancar. Sumberdaya yang dimaksud termasuk dana uang dan lain lain
- 3) **Design**; tahap ini memiliki kesamaan dengan tahap 2 (analisis sumberdaya) yaitu berhubungan dengan requirements, tapi tahap design ini lebih berfokus pada technical requirements seperti pemilihan bahasa pemrograman, arsitektur program, dan hardware yang dibutuhkan untuk menjalankan program.
- 4) **Implementation**; bagian implementasi tahap 1 sampe 3, dan sekarang productionnya jalan ajah.
- 5) **Testing** ; tahap setelah implementation, dan program sudah jadi, tinggal lihat atau ngetest apakah program sudah layak untuk costumer.
- 6) **Maintenance** ; tahap setelah program sudah di deploy ke costumer dan pengguna, dan tinggal fix bug, nambah fitur, perbaikan, dll

Source : <https://polluxintegra.co.id/waterfall-model-sdlc-apa-itu/>

- **Data Flow Diagram (DFD)**

Data Flow Diagram (DFD) adalah alat visual yang digunakan untuk menggambarkan alur informasi dalam sistem, mulai dari input hingga output. Diperkenalkan oleh Ed Yourdon dan Larry Constantine pada 1970, DFD berfungsi untuk menyampaikan rancangan sistem secara jelas, menggambarkan hubungan antar komponen dalam sistem, dan merancang model kerja yang sistematis. DFD membantu programmer dalam menganalisis sistem, merencanakan alur data, dan mengidentifikasi prioritas

dalam pengembangan perangkat lunak. Dengan menggunakan DFD, proses pengembangan sistem menjadi lebih terstruktur dan mudah dipahami.

Source : <https://it.telkomuniversity.ac.id/apa-itu-dfd-dan-fungsinya/>

- **Unified Modelling Language (UML)**

Unified Modeling Language (UML) adalah bahasa pemodelan yang berorientasi objek dan digunakan dalam pengembangan perangkat lunak untuk memastikan kualitas proses dan produk. UML membantu mendeskripsikan dan merancang sistem perangkat lunak, terutama yang dibangun dengan pemrograman berorientasi objek. UML mencakup berbagai jenis diagram, seperti use case diagram yang menggambarkan interaksi aktor dengan sistem, class diagram yang mempresentasikan kelas beserta atribut dan operasinya, serta activity diagram yang menggambarkan alur kerja atau aktivitas sistem. UML juga berperan dalam pemodelan sequence diagram, yang menjelaskan logika proses antara aktor dan sistem. Karena sifatnya yang fleksibel, UML dapat digunakan untuk berbagai metodologi, meskipun paling sering diterapkan dalam metodologi berbasis objek.

Source : <https://ejournal.unibba.ac.id/index.php/j-sika/article/download/839/694>

- **Entity Relationship Diagram (ERD)**

Entity Relationship Diagram (ERD) adalah diagram yang digunakan untuk merancang basis data, dengan menunjukkan hubungan antar entitas, atribut, dan relasinya. Entitas merupakan objek seperti mahasiswa, dosen, atau produk, sedangkan atribut menggambarkan karakteristik entitas tersebut, seperti NIM atau nama. ERD juga menampilkan kardinalitas, yang menentukan berapa banyak entitas yang dapat berelasi dengan entitas lainnya (1-1, 1-N, atau N-N). Pembuatan ERD melibatkan identifikasi entitas, atribut, dan relasi, sebelum akhirnya digambar menggunakan alat visualisasi seperti draw.io.

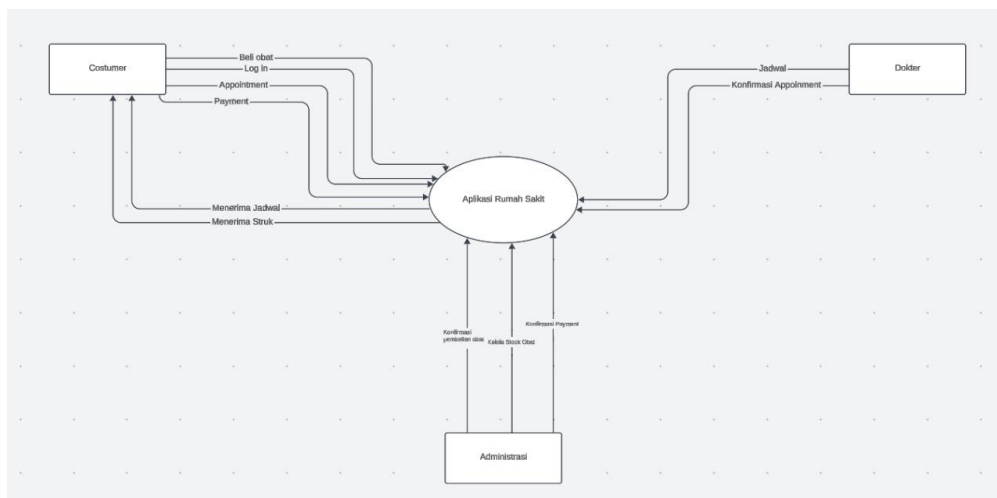
Source : <https://spada.uns.ac.id/mod/resource/view.php?id=185577>

BAB III

ANALISIS DAN PERANCANGAN

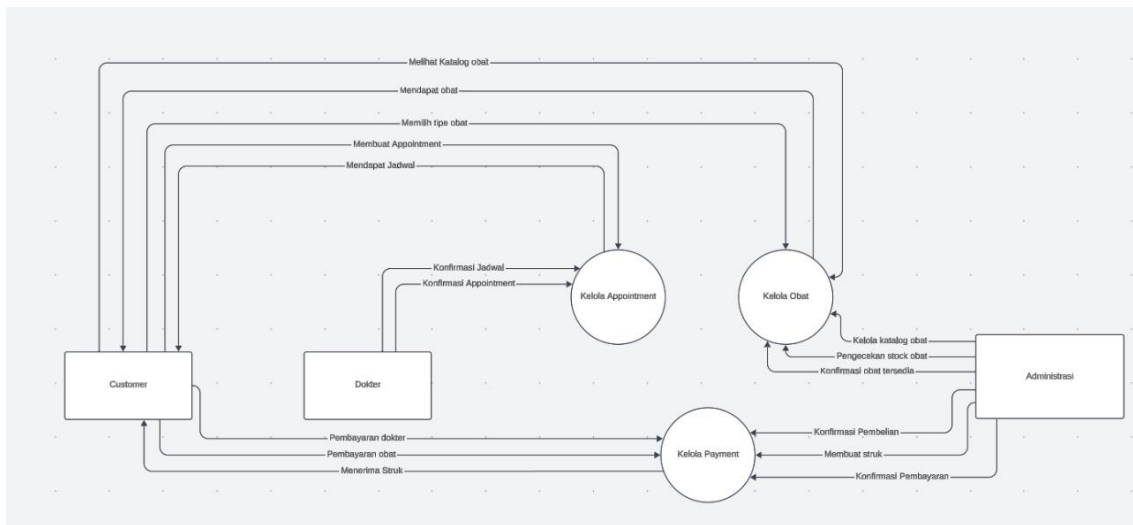
3.1 Desain Proses Terstruktur

- **Diagram Level 0**



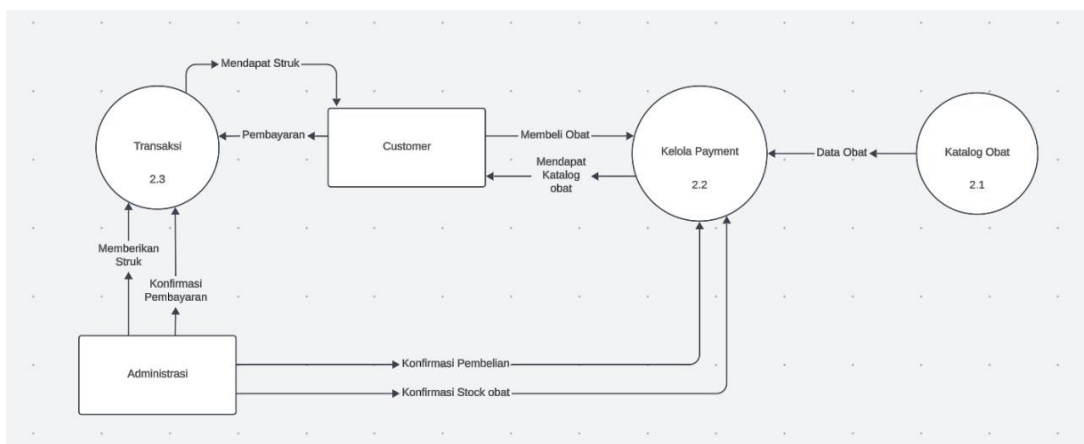
DFD Level 0 menggambarkan proses untuk sistem Aplikasi Rumah Sakit yang mengelola aliran data antara Customer, Dokter, dan Administrasi. Customer dapat melakukan beberapa aktivitas melalui aplikasi, seperti login, membeli obat, membuat appointment dengan dokter, dan melakukan pembayaran. Setelah appointment dibuat, customer akan menerima jadwal yang dikonfirmasi oleh dokter melalui aplikasi, serta menerima struk setelah pembayaran. Dokter mengirimkan jadwal dan konfirmasi appointment ke aplikasi, yang kemudian diteruskan ke customer. Aplikasi Rumah Sakit mengelola semua aktivitas ini, termasuk pengelolaan stok obat dan konfirmasi pembayaran, yang dikirimkan ke Administrasi. Administrasi kemudian mengelola data transaksi, termasuk konfirmasi pembelian obat dan stok yang tersedia. Dengan sistem ini, semua proses terkait pelayanan rumah sakit, mulai dari pembelian obat hingga appointment dokter, terkoordinasi melalui aplikasi terpusat yang terhubung dengan setiap entitas terkait.

- **Diagram Level 1**



DFD Level 1 menggambarkan proses alur sistem pengelolaan obat dan appointment dokter. Customer dapat melihat katalog obat, memilih obat, membuat appointment dengan dokter, dan melakukan pembayaran. Kelola Appointment mengelola jadwal appointment antara customer dan dokter, sementara Kelola Obat memastikan ketersediaan obat dengan mengelola katalog dan stok obat. Setelah itu, Kelola Payment menangani pembayaran obat dan layanan dokter, serta menghasilkan struk transaksi. Dokter mengonfirmasi appointment dan menerima pembayaran, sedangkan Administrasi mengelola konfirmasi pembayaran dan struk yang diterbitkan. Diagram ini menunjukkan bagaimana setiap proses saling berhubungan dalam sistem yang terintegrasi.

- **Diagram Level 2**



DFD Level 2 ini menggambarkan alur proses pemesanan obat, mulai dari customer memilih obat, melakukan pembayaran, hingga transaksi diselesaikan dengan penerimaan struk oleh customer. Administrasi juga memainkan peran penting dalam mengelola konfirmasi pembelian dan stok obat.

3.2 Desain OOP(UML)

- Use case

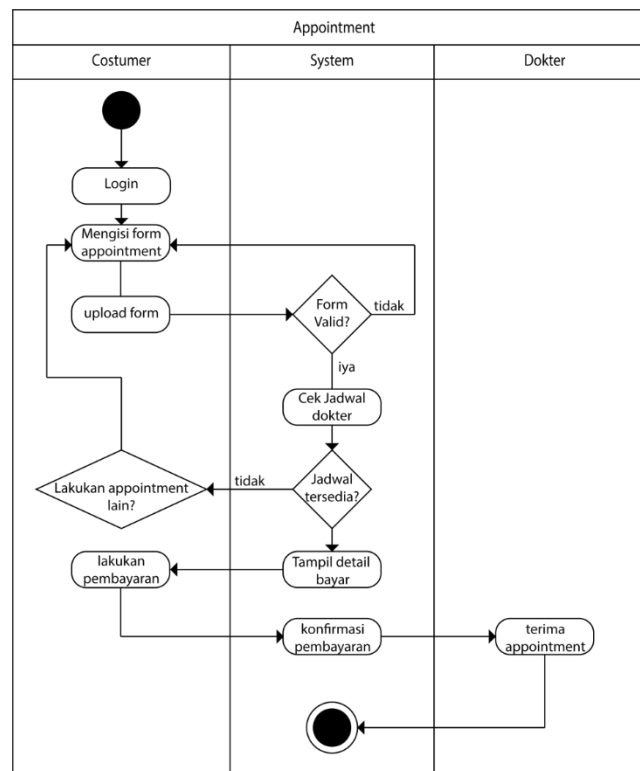


Diagram ini menggambarkan interaksi antara pasien, dokter dan administrasi dalam sistem rumah sakit. Pasien dapat melakukan berbagai aktifitas seperti membuat janji checkup (appointment), membeli obat. Untuk administrasi dapat mengelola pembelian dan stock obat, dan untuk dokter dapat melihat jadwal appointment nya.

- **Activity Diagram**

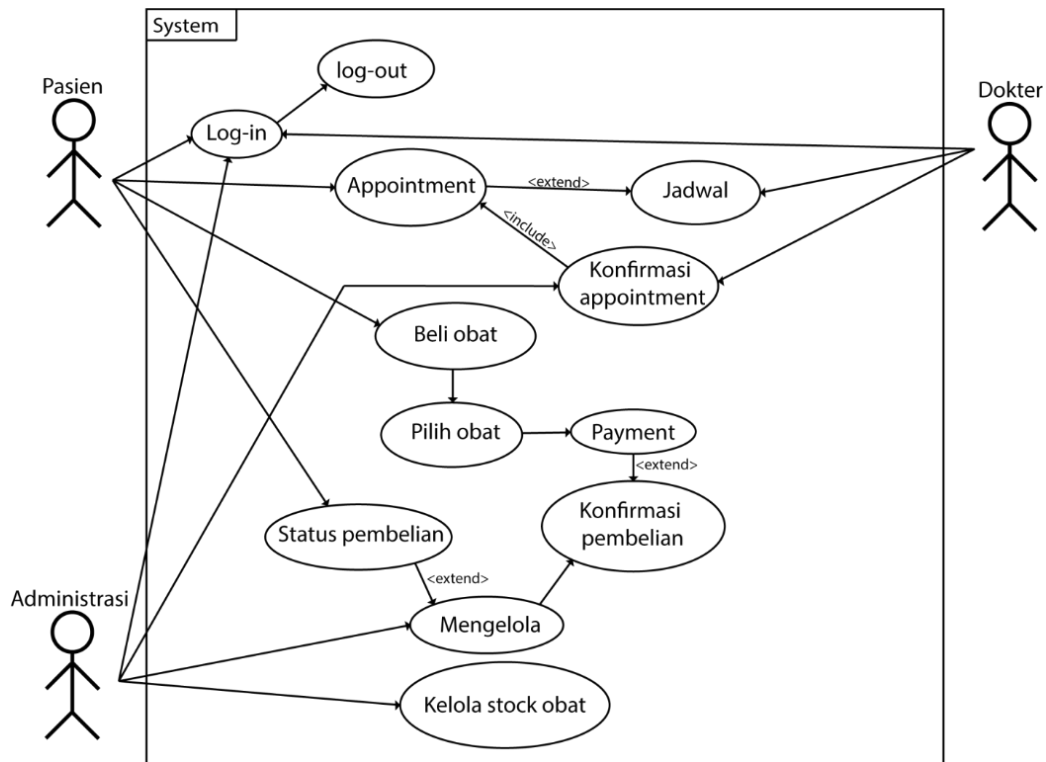


Diagram aktivitas ini menggambarkan alur langkah yang dilakukan oleh pasien untuk membuat janji pertemuan dengan dokter. Prosesnya dimulai dari pasien atau costumer melakukan login, kemudian pasien mengisi form permohonan appointment. Setelah formulir divalidasi, sistem akan memeriksa ketersediaan jadwal dokter yang dipilih. Jika jadwal tersedia, pasien dapat melanjutkan ke proses pembayaran. Setelah pembayaran terkonfirmasi, sistem akan mengirimkan notifikasi konfirmasi appointment kepada dokter. Dengan demikian, proses pembuatan appointment selesai.

- **Activity Diagram Pembelian Obat**

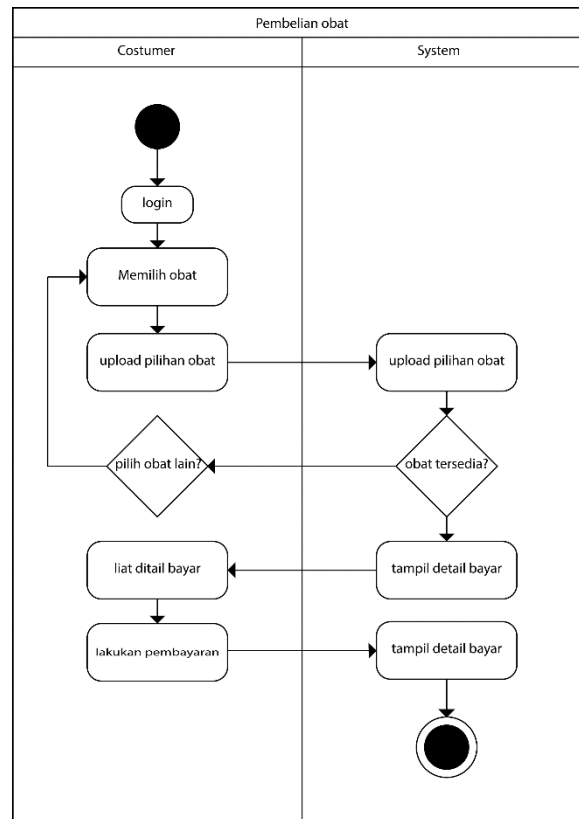


Diagram aktivitas ini menggambarkan alur langkah yang dilakukan oleh pelanggan untuk membeli obat. prosesnya dimulai dari pelanggan melakukan login, kemudian memilih obat yang diinginkan. setelah pilihan obat dikirimkan, sistem akan memeriksa ketersediaan stok obat yang di pilih. jika obat tersedia, pelanggan akan diberikan detail pembayaran dan melakukan transaksi. setelah pembayaran berhasil, proses pembelian selesai.

3.3 Desain ERD

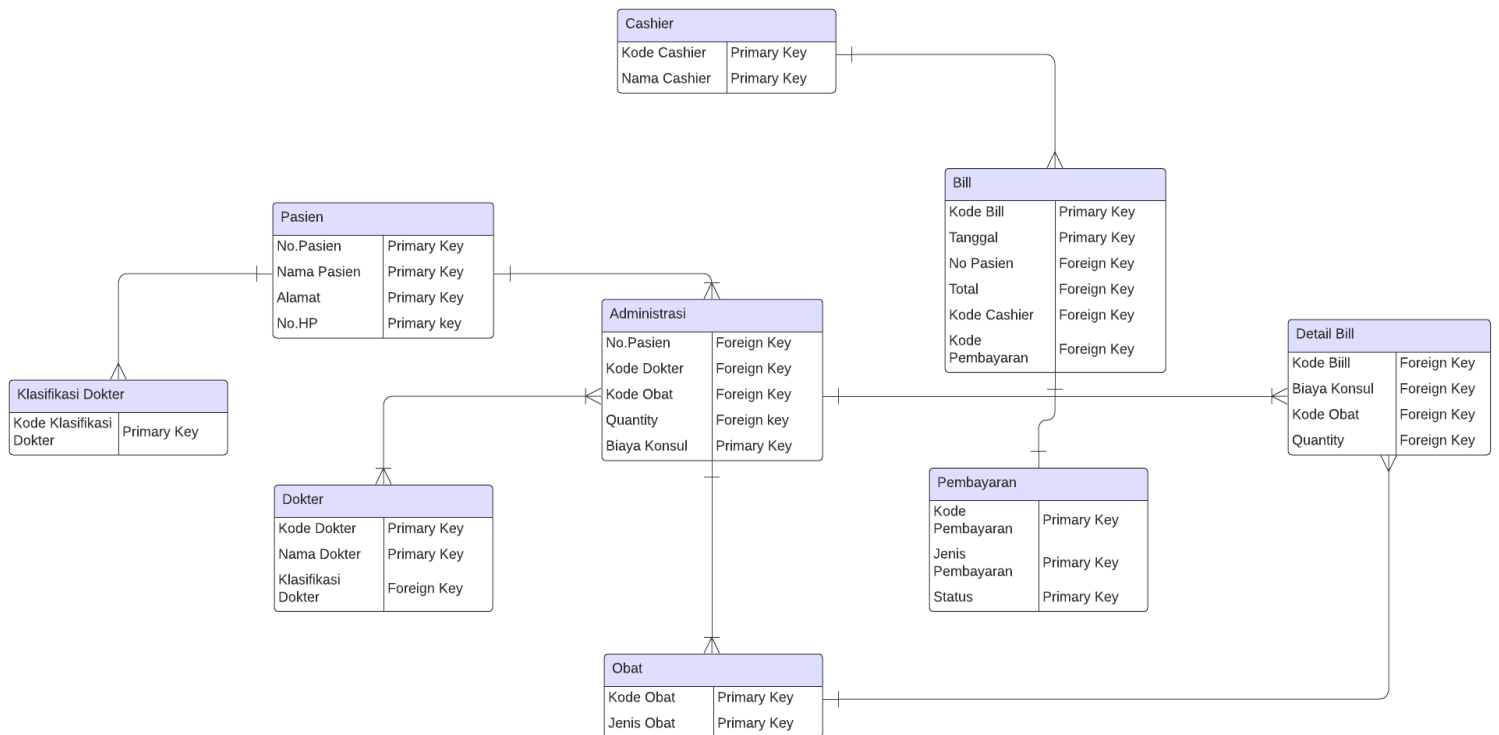


Diagram ERD ini menggambarkan sebuah sistem sederhana untuk apotek atau klinik. Sistem ini mencatat data tentang obat-obatan, pasien, dokter, dan transaksi penjualan obat. Relasi antar entitas menunjukkan bagaimana data-data tersebut saling terkait satu sama lain.

- **Entitas dan Atribut Utama:**

- a) **Pasien:** Mengandung informasi dasar tentang pasien seperti nomor pasien, nama, alamat, dan nomor telepon.
- b) **Dokter:** Menyimpan data tentang dokter, termasuk kode dokter, nama dokter, dan spesialisasi.
- c) **Obat:** Berisi informasi mengenai obat-obatan yang tersedia, seperti kode obat dan jenis obat.

- d) Kasir: Menyimpan data tentang kasir yang bertugas, seperti kode kasir dan nama kasir.
- e) Administrasi: Mencatat transaksi pendaftaran pasien, termasuk nomor bill, tanggal transaksi, nomor pasien, kode kasir, jumlah obat, biaya konsultasi, dan kode pembayaran.
- f) Detail Bill: Merinci item-item yang tercantum dalam satu bill, seperti kode bill, kode obat, jumlah obat, biaya konsultasi, dan biaya obat.
- g) Pembayaran: Menyimpan informasi tentang metode pembayaran yang digunakan, seperti kode pembayaran, jenis pembayaran, dan status pembayaran.
- h) Klasifikasi Dokter: Menentukan spesialisasi atau bidang keahlian dokter.

- **Relasi Antar Entitas:**

- a) Pasien dan Administrasi: Satu pasien dapat memiliki banyak transaksi (bill), tetapi satu transaksi hanya terkait dengan satu pasien.
- b) Dokter dan Administrasi: Satu dokter dapat menangani banyak pasien (dan dengan demikian banyak transaksi), dan satu transaksi dapat melibatkan satu atau lebih dokter.
- c) Obat dan Administrasi: Satu obat dapat muncul di banyak transaksi, tetapi satu transaksi dapat melibatkan banyak jenis obat.
- d) Kasir dan Administrasi: Satu kasir dapat menangani banyak transaksi, dan satu transaksi hanya ditangani oleh satu kasir.
- e) Detail Bill dan Administrasi: Satu bill dapat memiliki banyak detail item, tetapi satu detail item hanya terkait dengan satu bill.
- f) Pembayaran dan Administrasi: Satu transaksi (bill) hanya memiliki satu metode pembayaran.
- g) Dokter dan Klasifikasi Dokter: Satu dokter hanya memiliki satu spesialisasi.

No.	Nama	NIM	Pembagian Tugas
1.	Reynaldi	32230009	Latar Belakang, Teori umum, Desain ERD
2.	Varel Gabriel Wungow	32230062	Rumusan Masalah, Teori Khusus, Desain OOP (UML) use case
3.	Alexander Leon	32230004	Ruang lingkup/Batasan masalah, Teori Khusus, Desain terstruktur (DFD)
4.	Davin Cleon	32230082	Tujuan dan manfaat, Teori umum, Desain OOP (UML) Ability

DAFTAR PUSTAKA

- Amazon Web Services*. (n.d.). Retrieved from What is Systems Development Life Cycle (SDLC): <https://aws.amazon.com/id/what-is/sdlc/>
- Bithealth. (2024). Efisiensi operasional dan kepuasan pasien.
- Dibimbing. (2023). Pengertian sistem terintegrasi.
- Education, P. (2022). Principles of software engineering.
- Hill, M. (2023). Healthcare data management and analysis.
- O'Brien, J. A. (2021). *Introduction to information systems*. New York: McGraw-Hill Education.
- Pollux Integra*. (n.d.). Retrieved from Waterfall model SDLC: <https://polluxintegra.co.id/waterfall-model-sdlc-apa-itu/>
- RI, K. (2020). Pedoman sistem informasi manajemen rumah sakit (SIMRS).
- Sidiq, M. (2020). Rekayasa perangkat lunak.
- Spada. (2022)). Registrasi dalam layanan digital.
- Telkom University*. (n.d.). Retrieved from Apa itu DFD dan fungsinya: <https://it.telkomuniversity.ac.id/apa-itu-dfd-dan-fungsinya/>
- Unibba Journal*. (n.d.). Retrieved from Unified modeling language (UML): <https://ejournal.unibba.ac.id/index.php/j-sika/article/download/839/694>
- Universitas Sebelas Maret*. (n.d.). Retrieved from Entity relationship diagram (ERD): <https://spada.uns.ac.id/mod/resource/view.php?id=185577>
- Usaha, B. (2024). Operasional dalam perusahaan.
- WHO. (2021). Digital health interventions for health systems strengthening.