
Software Requirements Specification

for

CliniX Web-Based Application

Version 1.0 approved

Prepared by

Gharlan Winarno - 6181901044

Giullian Akmal - 6182101029

Metadia Yasinda Dula - 6182101031

Bram Matthew - 6182101043

Alexander Anova - 6182101050

Arlo Dante Hananvyasa - 6182201010

Parahyangan Catholic University

Wednesday, 25 September 2024

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Project Scope	1
1.5 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Features	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	3
2.5 Design and Implementation Constraints	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	5
3. System Features	5
3.1 Register dan Log in	5
3.2 Pendaftaran ke Dokter	6
3.3 Pemeriksaan Kondisi Awal Pasien	6
3.4 Pemeriksaan oleh Dokter dan Pemberian Resep	7
3.5 Pembayaran dan Cetak Tagihan	7
3.6 Pembayaran dan Cetak Tagihan	8
3.7 Manajemen Rekam Medis Pasien	8
3.8 Manajemen Jadwal Praktek Dokter dan Shift Perawat	9
4. External Interface Requirements	9
4.1 User Interfaces	9
4.2 Hardware Interfaces	10
4.3 Software Interfaces	10
4.4 Communications Interfaces	11
5. Other Nonfunctional Requirements	12
5.1 Performance Requirements	12
5.2 Safety Requirements	12
5.3 Security Requirements	13
5.4 Software Quality Attributes	13
6. Other Requirements	14
Appendix A: Glossary	14
Appendix B: Analysis Models	15
Appendix C: Issues List	

Revision History

Name	Date	Reason For Changes	Version
------	------	--------------------	---------

1. Introduction

1.1 Purpose

Tujuan pembuatan dokumen ini adalah untuk menganalisis, tentang sistem perangkat lunak CliniX. Proyek ini bertujuan untuk mengembangkan aplikasi berbasis web untuk meningkatkan layanan kepada pasien dan meningkatkan kinerja operasional CliniX. Persyaratan detail dari sistem perangkat lunak Clinix akan dijelaskan dalam dokumen ini.

1.2 Document Convention

Dokumen ini mengikuti Format MLA. Teks yang diberi cetak tebal digunakan untuk bagian judul dan sub-bagian. Teks yang dicetak miring digunakan untuk memberi label dan mengenali diagram.

1.3 Intended Audience and Reading Suggestions

Dokumen ini ditujukan untuk berbagai audiens yang terlibat dalam pengembangan dan penggunaan aplikasi pendaftaran klinik. Pengembang akan menggunakan dokumen ini untuk mengimplementasi fitur utama seperti pendaftaran pasien, mengatur pembayaran, merekam hasil pencatatan pasien dan dokter. Penguji akan memfokuskan perhatian pada pengujian fitur-fitur untuk memastikan semua nya benar dan sesuai spesifikasi. Pengguna akhir (pasien) akan menggunakan untuk mendaftar dan melihat jadwal dokter

1.4 Project Scope

Proyek ini bertujuan mengembangkan aplikasi pendaftaran klinik yang memungkinkan pasien mendaftar secara online, mengelola jadwal dokter, dan mengoptimalkan operasi klinik. Sistem ini akan meningkatkan pengalaman pasien dengan menyediakan akses mudah ke layanan medis dan meningkatkan efisiensi staf klinik. Fitur utama meliputi pendaftaran pasien online, pemesanan janji temu, manajemen rekam medis, dan pemrosesan pembayaran. Aplikasi ini akan mendukung tujuan klinik dalam menyediakan pelayanan kesehatan berkualitas tinggi dan mudah diakses oleh masyarakat.

1.5 References

IEEE Standard for Software Requirements Specifications (IEEE Std 830-1998), HL7 (Health Level Seven) Standards for Healthcare Information Exchange, Clinic internal policies and regulations

2. Overall Description

2.1 Product Perspective

Aplikasi Pendaftaran CliniX adalah sistem berbasis web yang dirancang untuk mengelola seluruh proses pendaftaran pasien, pemeriksaan medis, pencatatan diagnosa, dan proses pembayaran di poliklinik. Sistem ini akan menyediakan berbagai fitur bagi pengguna dengan peran yang berbeda, termasuk pasien, petugas administrasi, perawat, dan dokter. Tujuan dari aplikasi ini adalah untuk meningkatkan efisiensi dalam pengelolaan pelayanan kesehatan serta memberikan kemudahan akses informasi bagi semua pihak yang terlibat.

CliniX akan memiliki 4 tampilan muka. Masing-masing untuk Pasien, Dokter, Perawat, dan Petugas Administrasi. Petugas Administrasi dapat mendaftarkan pasien, melihat dan mengatur jadwal Dokter dan Perawat. Dokter dapat meng-submit hasil diagnosa dan resep obat untuk pasien. Perawat dapat mencatat informasi dasar pasien dan juga menginput informasi dasar pasien yang eksternal. Tampilan Muka untuk Pasien akan memberikan pilihan untuk melakukan registrasi, melihat dokter yang tersedia dan juga melihat dan menyelesaikan pembayaran.

2.2 Product Features

Berikut adalah fungsi utama yang dapat dilakukan oleh Sistem Informasi berbasis web CliniX.

- Pasien dapat melakukan registrasi secara online menggunakan formulir
- Pasien dapat melakukan booking untuk sesi pada klinik secara online.
- Pasien dapat melihat daftar jadwal praktek Dokter pada klinik.
- Pasien dapat melihat resep obat yang diberikan.
- Pasien dapat melihat tagihan dan dapat memilih metode pembayaran.
- Dokter dapat menginput hasil diagnosa pasien.
- Dokter dapat menginput resep obat untuk pasien.
- Perawat dapat mengisi dan mengupload formulir informasi dasar pasien.
- Perawat dapat mengupload hasil lab pasien yang berasal dari luar klinik.
- Petugas Administrasi dapat melihat dan mengatur jadwal praktek Dokter.
- Petugas Administrasi dapat melihat dan mengatur jadwal shift Perawat.
- Petugas Administrasi dapat memvalidasi booking yang dilakukan pasien.
- Petugas Administrasi dapat mendaftarkan pasien jika pasien belum memiliki nomor rekam medis.
- Petugas Administrasi dapat memberikan tagihan biaya terhadap pasien.
- Petugas Administrasi dapat melihat dan mengkonfirmasi bukti pembayaran pasien.
- Dapat menyimpan data Pasien, Dokter, Perawat dan Petugas Administrasi.
- Dapat menampilkan kuota pasien untuk masing-masing dokter yang praktek.

2.3 User Classes and Characteristics

Akan terdapat 4 tipe pengguna yang akan berinteraksi dengan sistem kami. Pertama akan ada Pasien. kemudian Petugas Administrasi, Perawat dan Dokter.

2.3.1 Pasien

Pasien akan berinteraksi langsung dengan sistem. Mereka akan registrasi online menggunakan website, mendaftar diri ke dokter yang tertentu, dan dapat melakukan pembayaran.

2.3.2 Petugas Administrasi

Petugas Administrasi akan bertugas untuk mendaftarkan pasien secara langsung jika pasien daftar dengan metode offline, mendaftar ulang pasien, mencatat pembayaran setiap pasien, dan membuat jadwal untuk dokter dan shift perawat

2.3.3 Perawat

Tugas Perawat adalah untuk mencatat informasi dasar pasien dan mengupload rekaman medis pasien yang berasal dari luar poliklinik

2.3.4 Dokter

Tugas Dokter adalah untuk mencatat hasil diagnosa dan mencatat obat-obatan yang diresepkan kepada pasien

2.4 Operating Environment

Aplikasi ini adalah aplikasi berbasis web yang berjalan di berbagai perangkat dan lokasi di CliniX. Berikut adalah gambaran lingkungan operasionalnya

2.4.1 Pasien/Keluarga Pasien

Pasien atau keluarganya dapat mengakses aplikasi melalui perangkat pribadi seperti smartphone, tablet, atau komputer desktop. Aplikasi akan berjalan di browser modern yang mendukung HTML5, CSS3, dan JavaScript.

2.4.2 Petugas Administrasi

Petugas administrasi menggunakan komputer desktop atau laptop yang tersedia di poliklinik untuk mengakses aplikasi. Aplikasi berjalan di browser dengan akses ke sistem manajemen pasien dan pendaftaran.

2.4.3 Perawat

Perawat menggunakan komputer desktop atau tablet yang terhubung dengan aplikasi untuk mencatat kondisi dasar pasien seperti tekanan darah, suhu tubuh, tinggi dan berat badan, serta keluhan pasien.

2.4.4 Dokter

Dokter menggunakan komputer desktop atau laptop di ruang praktek untuk mengakses aplikasi. Aplikasi ini digunakan untuk mencatat diagnosa penyakit, meresepkan obat, dan mengakses riwayat kesehatan pasien.

2.5 Design and Implementation Constraints

2.5.1 Operating System Constraints

Aplikasi harus kompatibel dengan semua sistem operasi desktop dan mobile modern, seperti Windows 10 ke atas, macOS, Android 8.0 ke atas, dan iOS 12 ke atas.

2.5.2 Browser Constraints

Aplikasi berbasis web harus dapat berjalan lancar pada browser modern seperti Google Chrome, Mozilla Firefox, Microsoft Edge, dan Safari versi terbaru.

Aplikasi harus mendukung HTML5 dan CSS3 dengan baik.
Dukungan untuk browser lama seperti Internet Explorer tidak diperlukan.

2.5.3 Device Constraints

Aplikasi harus responsif dan berjalan optimal pada perangkat desktop, tablet, dan smartphone. Fokus utama adalah pada perangkat desktop untuk petugas administrasi dan dokter, serta perangkat mobile (smartphone dan tablet) untuk pasien dan perawat.

2.5.4 Hardware Constraints

Perangkat server harus memiliki kapasitas penyimpanan dan memori yang cukup untuk menangani jumlah besar data pasien dan rekam medis.

2.6 User Documentation

Perangkat lunak ini akan disertai dengan materi pendukung berikut untuk membantu pengguna memahami dan menggunakan aplikasi:

- User Manual Versi 1.0: Panduan pengguna yang berisi langkah-langkah pendaftaran, penggunaan fitur-fitur utama aplikasi, serta cara melakukan pembayaran. Panduan ini akan tersedia dalam format PDF dan dapat diunduh dari situs poliklinik.
- Online Support: Dukungan online melalui situs resmi poliklinik di www.CliniX.com, di mana pengguna dapat mengakses FAQ, panduan pengguna tambahan, dan menghubungi tim dukungan pelanggan.

2.7 Assumptions and Dependencies

Asumsi:

- Semua pengguna memiliki akses ke perangkat dengan koneksi internet.
- Staf klinik memiliki keterampilan dasar penggunaan komputer.
- Volume pasien tidak akan melebihi 5000 per bulan.

Ketergantungan:

- Ketersediaan dan kinerja layanan pembayaran pihak ketiga.
- Pembaruan reguler sistem operasi dan perangkat lunak server.
- Kepatuhan berkelanjutan terhadap standar keamanan data kesehatan.

3. System Features

3.1 Register dan Log in

3.1.1 Description and Priority

Fitur ini membuat akun untuk pengguna baru dalam sistem. Agar pengguna dapat mendaftar ke dokter tertentu, ia harus memiliki akun dan harus masuk ke dalamnya. Fitur ini memasukkan detail pengguna ke dalam database. Ini adalah fitur yang penting sehingga memiliki prioritas yang tinggi.

Priority: High

3.1.2 Stimulus/Response Sequences

- **Stimulus:** Pasien mengklik tombol "Register" untuk memulai proses pendaftaran. Sistem menampilkan form untuk mengisi informasi pribadi pasien seperti nama lengkap, alamat, nomor telepon, alamat email, dan password.
- **Response:** Sistem memvalidasi informasi yang dimasukkan dan membuat akun baru bagi pasien. Jika validasi berhasil, sistem mengirimkan notifikasi pendaftaran sukses melalui email.
- **Stimulus:** Setelah mendaftar, pasien mengklik tombol "Login" dan memasukkan email dan password.
- **Response:** Sistem memverifikasi data login dan jika benar, pasien diarahkan ke halaman dashboard utama.

3.1.3 Functional Requirements

- REQ-1: Sistem harus menyediakan tombol untuk mendaftar (Register) dan login.
- REQ-2: Form data diri tempat pengguna mengisi data dirinya
- REQ-3: Database harus menyimpan tabel dengan detail semua pasien yang telah terdaftar.

3.2 Pendaftaran ke Dokter

3.2.1 Description and Priority

Fitur ini membuat akun untuk pengguna baru dalam sistem. Agar pengguna dapat mendaftar ke dokter tertentu, ia harus memiliki akun dan harus masuk ke dalamnya. Fitur ini memasukkan detail pengguna ke dalam database. Ini adalah fitur yang penting sehingga memiliki prioritas yang tinggi.

Priority: High

3.2.2 Stimulus/Response Sequences

- **Stimulus:** Pasien memilih dokter dari daftar dokter spesialis yang tersedia di halaman jadwal dokter, kemudian memilih opsi pendaftaran H-1 atau hari H.
- **Response:** Sistem mengecek ketersediaan kuota untuk dokter tersebut. Jika kuota tersedia, pasien berhasil didaftarkan dan notifikasi dikirimkan ke pasien.

3.2.3 Functional Requirements

- REQ-1: Sistem harus memungkinkan perawat untuk mencatat informasi dasar kondisi pasien sebelum pemeriksaan dokter.
- REQ-2: Data hasil pemeriksaan harus disimpan dalam rekam medis pasien dan dapat diakses oleh dokter.

3.3 Pemeriksaan Kondisi Awal Pasien

3.3.1 Description and Priority

Fitur ini memungkinkan perawat untuk mencatat informasi dasar kondisi pasien seperti tekanan darah, tinggi dan berat badan, suhu tubuh, serta keluhan pasien sebelum masuk ke ruangan dokter. Fitur ini penting untuk mempersiapkan dokter sebelum melakukan pemeriksaan.

Priority: High

3.3.2 Stimulus/Response Sequences

- **Stimulus:** Perawat memilih pasien yang telah mendaftar, kemudian mencatat hasil pemeriksaan awal seperti tekanan darah, tinggi, berat badan, suhu tubuh, dan keluhan yang dirasakan.
- **Response:** Sistem menyimpan data hasil pemeriksaan awal ke dalam rekam medis pasien.

3.3.3 Functional Requirements

- REQ-1: Sistem harus menampilkan jadwal dokter dengan pilihan untuk mendaftar di H-1 atau hari H.
- REQ-2: Sistem harus memvalidasi ketersediaan kuota untuk dokter yang dipilih sebelum menyelesaikan proses pendaftaran.
- REQ-3: Sistem harus menyimpan data pendaftaran di tabel yang sesuai.

3.4 Pemeriksaan oleh Dokter dan Pemberian Resep

3.4.1 Description and Priority

Dokter dapat memeriksa pasien sesuai urutan antrian dan mencatat diagnosa serta resep obat yang diberikan. Fitur ini sangat penting untuk memastikan riwayat medis pasien tersimpan dengan baik dan dapat diakses di masa mendatang.

Priority: High

3.4.2 Stimulus/Response Sequences

- **Stimulus:** Dokter memilih pasien dari daftar antrian dan melihat riwayat medis serta hasil pemeriksaan awal dari perawat.
- **Response:** Dokter mencatat diagnosa dan memberikan resep obat jika diperlukan, kemudian sistem menyimpan informasi tersebut ke rekam medis pasien.

3.4.3 Functional Requirements

- REQ-1: Sistem harus menyediakan antarmuka bagi dokter untuk melihat riwayat medis dan hasil pemeriksaan awal pasien.
- REQ-2: Sistem harus memungkinkan dokter mencatat diagnosa dan memberikan resep obat yang tercatat dalam rekam medis pasien.

3.5 Pembayaran dan Cetak Tagihan

3.5.1 Description and Priority

Fitur ini memungkinkan pasien untuk melakukan pembayaran baik secara online maupun di kasir poliklinik setelah selesai berobat. Setiap jenis layanan atau konsultasi dokter memiliki tarif yang berbeda-beda. Fitur ini penting untuk mengelola pembayaran pasien.

Priority: High

3.5.2 Stimulus/Response Sequences

- **Stimulus:** Pasien menyelesaikan konsultasi dengan dokter dan diarahkan ke bagian pembayaran. Pasien memilih metode pembayaran (online atau di kasir).
- **Response:** Sistem menampilkan rincian tagihan sesuai layanan yang diterima, dan setelah pembayaran selesai, sistem mencetak bukti pembayaran.

3.5.3 Functional Requirements

- REQ-1: Sistem harus menampilkan rincian biaya berdasarkan jenis layanan dan konsultasi dokter.

- REQ-2: Sistem harus mencatat pembayaran yang berhasil dan menyimpan bukti pembayaran dalam sistem.

3.6 Pembayaran dan Cetak Tagihan

3.6.1 Description and Priority

Fitur ini hanya tersedia untuk admin poliklinik. Admin dapat mengelola jadwal dokter dan mengatur detail terkait dokter yang bertugas di klinik.

Priority: Medium

3.6.2 Stimulus/Response Sequences

- **Stimulus:** Admin login dan memilih fitur untuk mengelola jadwal dokter. Admin memasukkan detail jadwal dokter dan menyimpannya.
- **Response:** Sistem memvalidasi input dan menyimpan jadwal dokter di database, serta menampilkan jadwal tersebut di halaman jadwal pasien.

3.6.3 Functional Requirements

- REQ-1: Sistem harus memungkinkan admin untuk menambah, mengubah, dan menghapus jadwal dokter.
- REQ-2: Sistem harus menyimpan data dokter dan jadwal yang terkait dalam database.

3.7 Manajemen Rekam Medis Pasien

3.7.1 Description and Priority

Fitur ini tersedia untuk perawat di poliklinik. Fitur ini memungkinkan perawat untuk meng-upload atau merekam informasi tambahan terkait rekam medis pasien yang tidak otomatis terekam di sistem (contoh: hasil lab dari luar poliklinik).

Priority: Medium

3.7.2 Stimulus/Response Sequences

- **Stimulus:** Perawat memilih fitur untuk mengelola rekam medis pasien. Perawat kemudian memasukkan data rekam medis baru atau meng-upload dokumen tambahan, seperti hasil lab dari luar poliklinik.
- **Response:** Sistem memvalidasi dan menyimpan data yang di-upload ke dalam rekam medis pasien, serta menampilkan data tersebut di halaman rekam medis untuk dokter.

3.7.3 Functional Requirements

- REQ-1: Sistem harus memungkinkan perawat untuk meng-upload atau mencatat informasi medis tambahan untuk pasien.
- REQ-2: Sistem harus memvalidasi dan menyimpan informasi rekam medis ke dalam database.
- REQ-3: Sistem harus mengizinkan perawat untuk melihat atau mengedit data rekam medis yang telah dimasukkan sebelum disimpan.

3.8 Manajemen Jadwal Praktek Dokter dan Shift Perawat

3.8.1 Description and Priority

Fitur ini tersedia untuk admin di poliklinik. Admin dapat mengelola jadwal praktek dokter serta jadwal shift perawat. Jadwal ini akan ditampilkan kepada perawat dan pasien yang mengakses sistem.

Priority: **Medium**

3.8.2 Stimulus/Response Sequences

Stimulus: Admin login dan memilih fitur manajemen jadwal. Admin memasukkan jadwal praktek dokter dan shift perawat, kemudian menyimpan data.

Response: Sistem memvalidasi jadwal yang dimasukkan dan menyimpannya di database. Jadwal yang valid akan ditampilkan di halaman dokter dan perawat, serta di halaman jadwal untuk pasien.

3.8.3 Functional Requirements

REQ-1: Sistem harus memungkinkan admin untuk menambah, mengubah, dan menghapus jadwal praktek dokter.

REQ-2: Sistem harus memungkinkan admin untuk menambah, mengubah, dan menghapus jadwal shift perawat.

REQ-3: Sistem harus menampilkan jadwal praktek dokter kepada pasien.

REQ-4: Sistem harus menampilkan jadwal shift perawat kepada perawat yang bersangkutan.

4 External Interface Requirements

4.1 Member Interface

1. Log-In and Register Page

Saat anggota pertama kali memasukkan alamat web aplikasi ke peramban, anggota akan disambut oleh antarmuka Log-In dan Register. Anggota yang sudah terdaftar dapat masuk dengan memasukkan email dan kata sandi akun mereka, sementara anggota yang belum memiliki akun dapat menekan tombol "Register" untuk membuat akun baru.

2. Member page

Halaman ini berfungsi sebagai antarmuka utama bagi anggota untuk mengakses berbagai layanan yang disediakan oleh aplikasi, yaitu memesan janji temu dan mengakses informasi setelah janji temu seperti memeriksa total biaya dan melihat resep.

4.2 Hardware Interfaces

Sistem ini akan berinteraksi dengan beberapa perangkat keras di CliniX untuk mendukung operasional klinik, seperti pendaftaran, pencatatan medis, dan pembayaran. Berikut adalah karakteristik logis dan fisik dari antarmuka perangkat keras yang didukung:

Printer Struk Pembayaran:

Sistem akan terhubung dengan printer struk untuk mencetak rincian pembayaran setelah pasien menyelesaikan layanan medis di poliklinik. Printer ini biasanya merupakan thermal printer yang mendukung antarmuka USB atau jaringan lokal (LAN). Data yang dikirim ke printer termasuk rincian layanan, biaya, dan informasi pasien.

Kartu Pembayaran (Credit/Debit Card Reader):

Sistem mendukung pembayaran melalui kartu kredit/debit dengan menggunakan perangkat pembaca kartu yang terpisah. Interaksi dengan perangkat ini tidak langsung terhubung dengan sistem, tetapi hasil pembayarannya dimasukkan secara manual oleh petugas administrasi ke dalam sistem setelah pembayaran berhasil.

Komputer Desktop dan Tablet:

Petugas administrasi, perawat, dan dokter akan menggunakan komputer desktop atau tablet untuk mengakses sistem. Sistem harus kompatibel dengan perangkat keras ini, dan antarmuka perangkat input seperti keyboard, mouse, atau layar sentuh harus berfungsi dengan baik.

4.3 Software Interfaces

Sistem ini akan terhubung dengan beberapa komponen perangkat lunak lain untuk memastikan kelancaran operasional dan integrasi dengan komponen eksternal. Berikut adalah antarmuka perangkat lunak yang akan digunakan:

Database:

SQL Server Management Studio (versi terbaru) akan digunakan sebagai sistem manajemen basis data relasional untuk menyimpan data pasien, riwayat kesehatan, diagnosa, resep, dan pembayaran.

Sistem akan berkomunikasi dengan database melalui API **RESTful** atau **GraphQL** untuk CRUD (Create, Read, Update, Delete) operasi data.

Data yang akan dibagikan antara komponen meliputi informasi pasien, riwayat rekam medis, informasi dokter, jadwal praktek, serta informasi pembayaran.

Sistem Operasi:

Linux (Ubuntu atau CentOS) atau Windows Server akan menjadi sistem operasi yang mendukung aplikasi server backend. Aplikasi client-side akan dijalankan di browser yang berjalan pada sistem operasi desktop seperti Windows 10/11, macOS, serta Android dan iOS pada perangkat mobile.

API:

Sistem ini akan terhubung dengan API eksternal untuk layanan tertentu seperti integrasi dengan laboratorium pihak ketiga atau sistem pembayaran online.

Data yang dikirim dan diterima melalui API mencakup informasi laboratorium (hasil tes) dan transaksi pembayaran. Protokol yang digunakan untuk komunikasi adalah HTTPS untuk menjaga keamanan data yang ditransmisikan.

Libraries/Frameworks:

Backend akan dibangun menggunakan **Java Spring** untuk menangani permintaan dari client, memproses data, dan berkomunikasi dengan database.

Frontend menggunakan **HTML** dan **CSS** untuk membangun antarmuka pengguna yang interaktif dan responsif. Axios atau Fetch API akan digunakan di frontend untuk berkomunikasi dengan backend melalui HTTP requests.

4.4 Communications Interfaces

Aplikasi web untuk CliniX akan berkomunikasi dengan berbagai layanan dan komponen eksternal maupun internal menggunakan protokol komunikasi dan standar berikut:

Protokol Komunikasi:

HTTP/HTTPS: Komunikasi antara aplikasi web dan server backend akan menggunakan protokol HTTPS untuk memastikan keamanan dan enkripsi data selama transmisi. Semua komunikasi dengan API eksternal (misalnya, layanan laboratorium pihak ketiga atau gateway pembayaran) juga akan melalui HTTPS untuk menjaga kerahasiaan dan integritas data.

Web Browser Communication:

Aplikasi web akan berjalan pada browser modern seperti Google Chrome, Mozilla Firefox, Microsoft Edge, dan Safari. Komunikasi antara browser dan server akan mengikuti standar RESTful API yang menggunakan JSON (JavaScript Object Notation) sebagai format data untuk memfasilitasi komunikasi yang ringan dan efisien.

WebSockets dapat digunakan untuk fitur yang memerlukan komunikasi real-time, seperti pembaruan antrian pasien secara langsung atau notifikasi langsung dari dokter atau petugas.

Network Communication:

Sistem akan berkomunikasi dengan server melalui jaringan lokal (LAN) atau internet. Semua data sensitif, seperti informasi pasien dan transaksi pembayaran, akan dienkripsi dengan protokol TLS (Transport Layer Security) untuk memastikan keamanan selama pengiriman data melalui jaringan.

Kecepatan transfer data akan bergantung pada bandwidth jaringan, tetapi sistem dioptimalkan untuk bekerja dalam koneksi jaringan kecepatan menengah hingga tinggi.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- **System Response Time:** Sistem harus merespon tindakan pengguna (misalnya, memuat jadwal dokter, memesan janji temu, mengakses rekam medis pasien) dalam waktu kurang dari 2 detik dalam kondisi normal.
- **Scalability:** Sistem harus mendukung hingga 500 pengguna secara bersamaan tanpa penurunan kinerja yang signifikan, serta mengakomodasi pertumbuhan klinik di masa depan.
- **Data Processing:** Sistem harus mampu menangani pemrosesan batch untuk janji temu, resep, dan penagihan secara tepat waktu, memastikan semua catatan diperbarui dalam waktu 5 menit setelah transaksi selesai.
- **Availability:** Sistem harus memiliki uptime sebesar 99.9% untuk memastikan akses yang terus-menerus bagi pasien dan staf, terutama selama jam operasional klinik.

5.2 Safety Requirements

- **Data Integrity:** Rekam medis, resep, dan data penagihan harus dilindungi dari akses, modifikasi, atau kehilangan yang tidak sah. Backup data secara rutin harus dilakukan setiap 24 jam, dan pengujian pemulihan data harus dilakukan setiap kuartal.
- **Error Handling:** Sistem harus menampilkan pesan kesalahan yang informatif untuk input yang salah atau malfungsi sistem, sehingga pengguna dapat memahami masalah dan mengetahui langkah selanjutnya.
- **Redundancy:** Fungsi sistem yang kritis, seperti penjadwalan janji temu dan manajemen data pasien, harus memiliki redundansi untuk memastikan ketersediaan selama kegagalan sistem parsial.

5.3 Security Requirements

User Authentication: Semua pengguna (admin, dokter, perawat, dan pasien) harus melakukan autentikasi menggunakan kredensial unik. **Multi-factor authentication (MFA)** harus diterapkan untuk admin, dokter, dan perawat.

Access Control: Sistem harus memiliki **role-based access control (RBAC)**, di mana:

- Pasien hanya dapat mengakses data mereka sendiri dan layanan terkait (janji temu, tagihan, resep).
- Admin dapat mengakses semua fungsi sistem tetapi dibatasi dari melihat rincian rekam medis.
- Dokter dapat mengakses dan memperbarui riwayat medis pasien, hasil tes, diagnosis, dan resep.
- Perawat dapat melihat dan memperbarui informasi umum pasien serta hasil tes, tetapi tidak dapat mengakses diagnosis atau resep.

Data Encryption: Data pasien yang sensitif, termasuk rekam medis dan informasi tagihan, harus dienkripsi baik saat disimpan maupun selama transmisi menggunakan protokol standar industri (misalnya, **AES-256** untuk penyimpanan dan **TLS** untuk transmisi).

Audit Trail: Sistem harus menjaga log terperinci dari semua akses dan modifikasi sistem, terutama yang berkaitan dengan data pasien. Log ini harus disimpan minimal selama 5 tahun dan tersedia untuk audit.

5.4 Software Quality Attributes

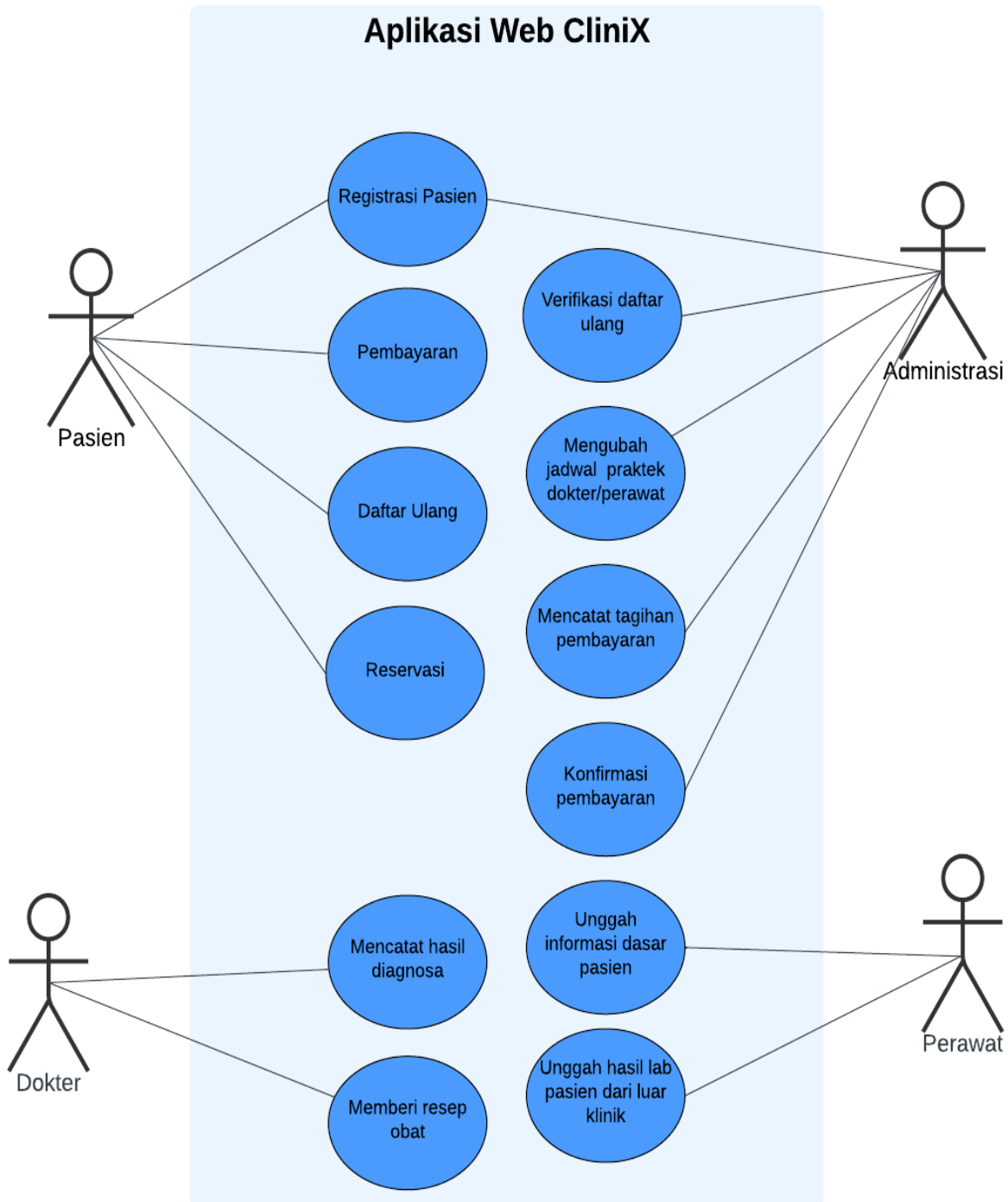
- **Reliabilitas:** Sistem harus memastikan tingkat reliabilitas yang tinggi, meminimalkan waktu tidak aktif, dan menjaga kinerja yang konsisten di seluruh fitur. Jika terjadi kegagalan, sistem harus pulih secara otomatis dalam waktu 10 menit.
- **Kegunaan:** Sistem harus menyediakan **user-friendly GUI** yang intuitif dengan navigasi yang jelas untuk setiap kelompok pengguna. Umpan balik dari pengguna (terutama dari pasien) akan dikumpulkan secara berkala untuk meningkatkan pengalaman.
- **Pemeliharaan:** Arsitektur sistem harus memungkinkan pembaruan dan perbaikan bug dengan mudah. **Modular design** akan memungkinkan peningkatan fitur tertentu tanpa mempengaruhi seluruh sistem.
- **Portabilitas:** Sistem harus kompatibel dengan peramban web umum (misalnya, Chrome, Firefox, Edge) dan dapat diakses di perangkat desktop maupun mobile dengan fungsionalitas penuh.
- **Skalabilitas:** Sistem harus dapat mengakomodasi peningkatan dalam catatan pasien, interaksi pengguna, dan cabang klinik tanpa memerlukan konfigurasi ulang besar atau menghadapi kemacetan kinerja.

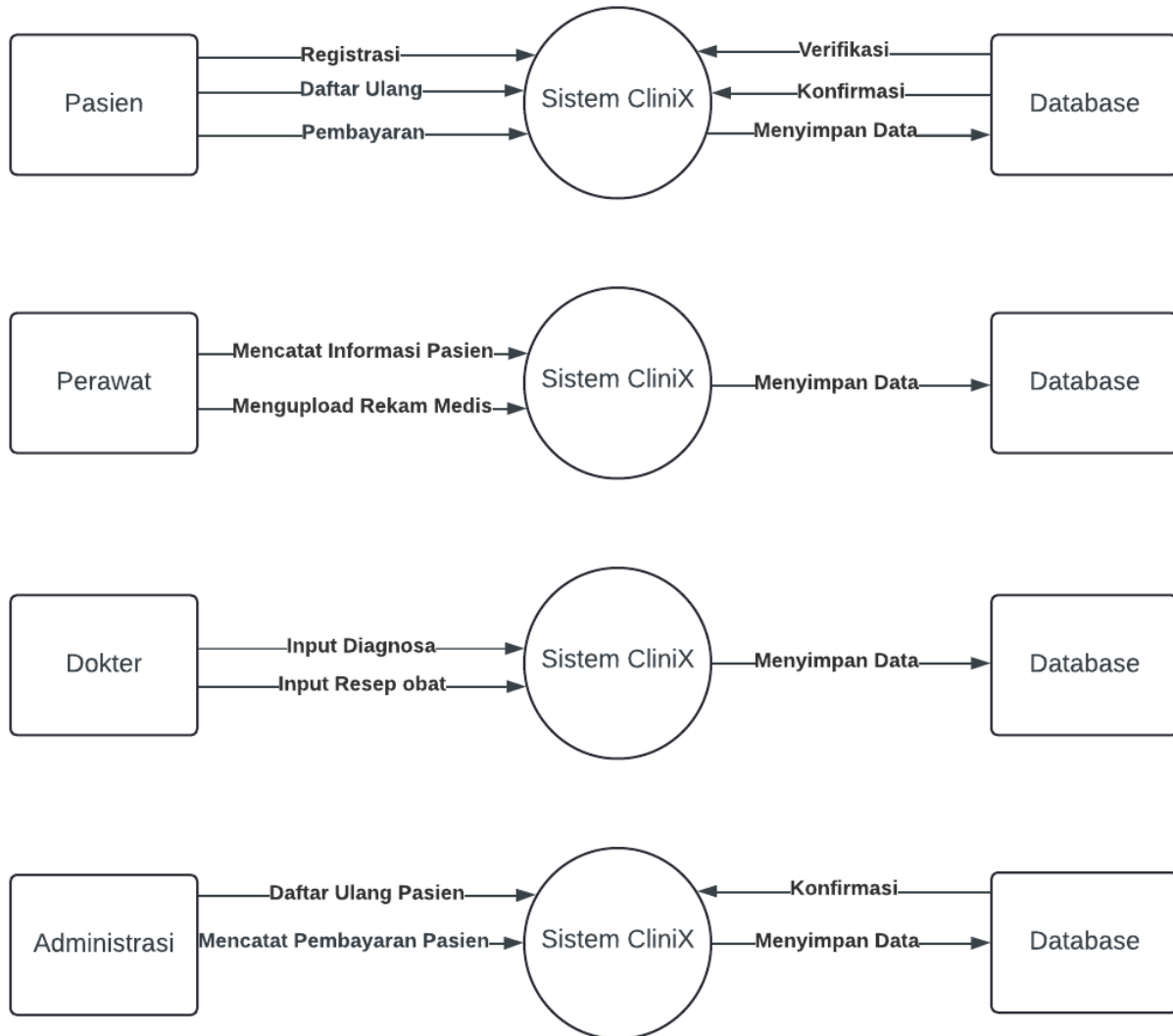
Appendix A: Glossary

1. **API (Application Programming Interface):** Sekumpulan aturan dan alat untuk membangun aplikasi perangkat lunak, memungkinkan sistem yang berbeda untuk berkomunikasi satu sama lain.
2. **Otentikasi:** Proses untuk memverifikasi identitas pengguna sebelum memberikan akses ke sistem atau layanan.
3. **CRUD (Create, Read, Update, Delete):** Operasi dasar yang dilakukan pada data dalam basis data, meliputi pembuatan, pengambilan, modifikasi, dan penghapusan catatan.
4. **Dokter:** Kelas pengguna dalam sistem yang bertugas mendiagnosis pasien dan meresepkan obat.
5. **Enkripsi:** Metode pengamanan data dengan mengubahnya menjadi kode untuk mencegah akses yang tidak sah.
6. **Front-End:** Bagian dari aplikasi yang berinteraksi langsung dengan pengguna, biasanya melibatkan antarmuka pengguna.
7. **HTTPS (Hypertext Transfer Protocol Secure):** Versi aman dari HTTP yang mengenkripsi data selama transmisi di web.
8. **Rekam Medis:** Catatan yang berisi informasi pasien, diagnosa, perawatan, dan resep dalam sistem.
9. **Pasien:** Kelas pengguna utama dalam sistem yang melakukan registrasi, memesan janji temu, dan melihat resep.
10. **Resep:** Perintah tertulis dari dokter untuk obat yang diberikan kepada pasien setelah diagnosis.
11. **Redundansi:** Penyertaan komponen atau cadangan tambahan untuk memastikan bahwa operasi sistem dapat terus berjalan jika terjadi kegagalan.
12. **Server:** Komputer atau perangkat lunak yang mengelola akses ke sumber daya atau layanan terpusat dalam aplikasi CliniX.
13. **Otentikasi Pengguna:** Langkah keamanan untuk memverifikasi identitas pengguna sebelum memberikan akses ke sistem, biasanya melalui kredensial seperti nama pengguna dan kata sandi.

Appendix B: Analysis Models

B1 : Use Case Diagram



B2 : DFD Level 0

B3 : Entity Relationship Diagram

