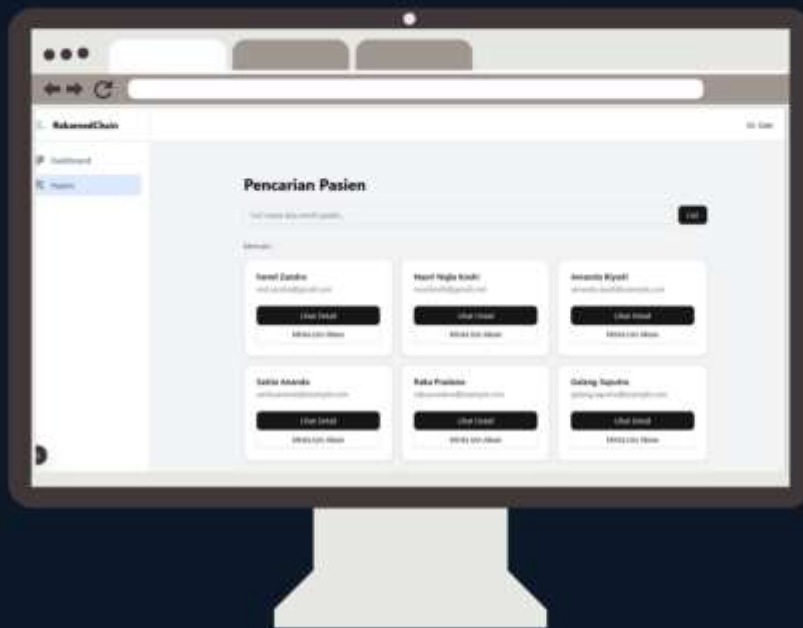




# DOKUMENTASI REKAMEDCHAIN

Sebuah Tinjauan Teknis dan Strategis Pengembangan MVP v1.0



Disusun oleh Trifur Labs  
2025

## DAFTAR ISI

DAFTAR ISI .....	i
DAFTAR GAMBAR .....	iv
DAFTAR TABEL .....	iv
1. Deskripsi Proyek .....	1
1.1. Latar Belakang .....	1
1.2. Tujuan Pengembangan Aplikasi .....	1
1.3. Solusi yang Ditawarkan.....	2
1.3.1. Lapisan Data & Skalabilitas (Off-Chain Storage).....	2
1.3.2. Lapisan Kedaulatan Pasien (Self-Sovereign Identity - SSI) .....	2
1.3.3. Lapisan Konsensus & Integritas (Blockchain) .....	2
1.4. Target Pengguna .....	2
1.4.1. Pasien.....	2
1.4.2. Tenaga Medis (Dokter) .....	3
1.5. Ringkasan Teknologi yang Digunakan .....	3
1.5.1. Frontend (Aplikasi Pasien) .....	3
1.5.2. Frontend (Portal Dokter) .....	3
1.5.3. Backend.....	3
1.5.4. Database & Penyimpanan .....	3
1.5.5. Blockchain.....	4
1.5.6. Infrastruktur & DevOps .....	4
1.6. Kesimpulan Singkat .....	4
2. Dokumentasi Fitur dan Implementasi .....	4
2.1. Autentikasi & Manajemen Peran .....	4
2.2. Alur Rekam Medis & Data .....	6
2.3. Kedaulatan Pasien & Keamanan Data.....	7
2.4. Transparansi & Jejak Audit .....	9
2.5. Fitur yang Direncanakan (Menuju Skala Penuh) .....	9
2.6. Kesimpulan.....	10
3. Arsitektur Sistem: RekamedChain.....	10
3.1. Penjelasan Singkat Alur Sistem.....	10
3.1.4. Interaksi dengan Lapisan Data .....	11
3.1.5. Respon ke Frontend .....	11
3.2. Daftar Teknologi dan Framework.....	11
3.2.1. Frontend (Aplikasi Pasien) .....	11
3.2.2. Frontend (Portal Dokter) .....	11

3.2.3.	Backend .....	11
3.2.4.	Database & Penyimpanan .....	12
3.2.5.	Blockchain.....	12
3.2.6.	Infrastruktur & DevOps .....	12
3.3.	Diagram Arsitektur Sistem.....	12
3.4.	Penjelasan Hubungan Antar Komponen .....	12
3.4.1.	Frontend → Backend.....	12
3.4.2.	Backend → PostgreSQL .....	13
3.4.3.	Backend → IPFS .....	13
3.4.4.	Backend → Hardhat .....	13
3.5.	Ringkasan Keamanan dan Otentikasi .....	13
3.5.1.	Autentikasi Sesi.....	13
3.5.2.	Otorisasi Berbasis Peran (RBAC) .....	13
3.5.3.	Self-Sovereign Identity (SSI).....	13
3.5.4.	Enkripsi Data At-Rest .....	13
3.5.5.	Integritas Data .....	13
4.	Database Design .....	14
4.1.	Penjelasan Struktur Database yang Digunakan .....	14
4.2.	Daftar Tabel dan Atribut .....	14
4.2.1.	Tabel Users.....	14
4.2.2.	Tabel MedicalRecords .....	15
4.2.3.	Tabel ConsentRequests .....	15
4.2.4.	Tabel BlockchainLedger .....	16
4.3.	Diagram ERD .....	17
4.4.	Penjelasan Relasi Antar Tabel .....	17
4.4.1.	Relasi Utama.....	17
4.5.	Alasan Perancangan.....	18
5.	Use Case Diagram dan Deskripsi: RekamedChain .....	19
5.1.	Daftar Aktor .....	19
5.2.	Diagram Use Case .....	20
5.2.1.	Diagram Use Case Pasien .....	20
5.2.2.	Diagram Use Case Tenaga Medis (Dokter).....	21
5.3.	Daftar dan Deskripsi Use Case .....	21
5.3.1.	Aktor: Pasien .....	21
5.3.2.	Aktor: Tenaga Medis (Dokter) .....	23
6.	Desain UI/UX: RekamedChain .....	25

6.1.	Makna Logo.....	25
6.2.	Prinsip Desain dan Gaya Visual .....	25
6.3.	Alur Navigasi (Flow).....	26
6.4.	Deskripsi Halaman Utama dan Komponennya .....	26
6.4.1.	Aplikasi Mobile (Pasien) .....	26
6.4.2.	Portal Web (Dokter) .....	27
6.5.	Bagaimana UI Mendukung UX.....	27
7.	Rencana Pengembangan: RekamedChain.....	27
7.1.	Evaluasi Versi MVP .....	27
7.1.1.	Kelebihan & Pencapaian Utama.....	27
7.1.2.	Kendala & Area Peningkatan .....	28
7.2.	Daftar Fitur Lanjutan yang Direncanakan.....	28
7.3.	Roadmap Pengembangan dan Prioritas .....	28
7.3.1.	Tahap 1: Penguatan Fondasi & UX .....	28
7.3.2.	Tahap 2: Implementasi Kedaulatan Penuh & Interaktivitas .....	29
7.3.3.	Tahap 3: Keamanan Lanjutan & Visi Jangka Panjang.....	29
7.4.	Potensi Ekspansi .....	29
8.	Kontribusi Tim dan Pembagian Peran: Tim Trifur.....	29
8.1.	Daftar Anggota Tim.....	30
8.2.	Peran dan Tanggung Jawab .....	30
8.3.	Deskripsi Kontribusi Selama Pengembangan .....	31
8.3.1.	Farrel Zandra .....	31
8.3.2.	Reqi Jumentara Hapid .....	31
8.3.3.	Umar Faruq Robbany .....	31
8.4.	Tools Kolaborasi.....	32
8.5.	Kesimpulan Kerja Tim .....	32

## DAFTAR GAMBAR

Gambar 1 Halaman Registrasi Portal Web .....	5
Gambar 2 Halaman Login Portal Web .....	6
Gambar 3 Tambah Rekam Medis Baru .....	7
Gambar 4 Riwayat Medis Pasien .....	7
Gambar 5 Manajemen Izin Akses .....	8
Gambar 6 Simulasi Blockchain (Lihat Ledger) .....	9
Gambar 7 Diagram Arsitektur Sistem .....	12
Gambar 8 ERD untuk MVP saat ini.....	17
Gambar 9 Use Case Diagram Pasien.....	20
Gambar 10 Use Case Diagram Dokter .....	21
Gambar 11 Logo RekamedChain .....	25
Gambar 12 Anggota Trifur .....	30

## DAFTAR TABEL

Tabel 1 Rincian Fitur Autentikasi dan Manajemen Peran .....	4
Tabel 2 Rincian Fitur Rekam Medis dan Data .....	6
Tabel 3 Rincian Fitur Autentikasi dan Manajemen Peran .....	7
Tabel 4 Rincian Fitur Transparansi & Jejak Audit .....	9
Tabel 5 Fitur yang Direncanakan.....	9
Tabel 6 Tabel Users.....	14
Tabel 7 Tabel MedicalRecords .....	15
Tabel 8 Tabel ConsentRequests .....	15
Tabel 9 Tabel BlockchainLedger .....	16
Tabel 10 Deskripsi Use Case Pasien .....	21
Tabel 11 Deskripsi Use Case Dokter .....	23
Tabel 12 Peran dan Tanggung Jawab.....	30

## 1. Deskripsi Proyek

**Nama** : RekamedChain

**Versi** : 1.0 (MVP)

**Tanggal** : 9 Oktober 2025

### 1.1. Latar Belakang

Sistem kesehatan digital di Indonesia menghadapi tiga tantangan sistemik yang krusial: fragmentasi data antar fasilitas kesehatan (faskes), kerentanan keamanan siber terhadap data medis yang sensitif, dan ketiadaan kedaulatan pasien atas data kesehatan pribadi mereka.

Meskipun regulasi seperti *Permenkes No. 24 Tahun 2022 tentang Rekam Medis Elektronik (RME)* dan platform SATUSEHAT telah mendorong digitalisasi, fondasi teknis yang ada masih berbasis arsitektur terpusat yang rapuh. Hal ini menghambat interoperabilitas data, menurunkan kepercayaan publik, dan menempatkan pasien pada posisi yang lemah dalam mengontrol data mereka sendiri.

Proyek RekamedChain dikembangkan untuk menjawab tantangan fundamental ini dengan merombak paradigma pengelolaan rekam medis.

### 1.2. Tujuan Pengembangan Aplikasi

Pengembangan RekamedChain bertujuan untuk menghasilkan sebuah Minimum Viable Product (MVP) yang fungsional dengan tujuan strategis sebagai berikut:

#### 1. Membangun Arsitektur Terdesentralisasi

Menciptakan purwarupa sistem rekam medis yang menjamin keamanan, integritas, dan interoperabilitas data dengan memanfaatkan teknologi blockchain dan penyimpanan terdistribusi.

#### 2. Mewujudkan Kedaulatan Pasien (Self-Sovereign Identity - SSI)

Mengimplementasikan model SSI untuk memberikan pasien kontrol penuh dalam mengelola dan memberikan persetujuan akses terhadap data medis pribadi mereka secara kriptografis.

#### 3. Membuktikan Kelayakan Konsep (Proof-of-Concept)

Menyediakan bukti konsep yang solid bahwa arsitektur hibrida (on-chain dan off-chain) dapat diimplementasikan secara efektif untuk kasus penggunaan rekam medis.

#### 4. Menyiapkan Fondasi untuk Interoperabilitas Nasional

Membangun fondasi sistem yang dirancang untuk dapat diintegrasikan dengan ekosistem kesehatan nasional seperti platform SATUSEHAT.

### 1.3. Solusi yang Ditawarkan

RekamedChain menawarkan solusi inovatif melalui arsitektur tiga lapis yang bekerja secara sinergis:

#### 1.3.1. Lapisan Data & Skalabilitas (Off-Chain Storage)

Lapisan ini terdiri dari beberapa komponen, yakni sebagai berikut:

- 1) Data rekam medis aktual (diagnosis, catatan) dienkrpsi dengan **AES-256** sebelum disimpan di **PostgreSQL**.
- 2) File lampiran (hasil lab, radiologi, dll) disimpan secara **terdesentralisasi** di **IPFS**.
- 3) Pendekatan ini menjamin **kerahasiaan data at-rest** sekaligus menjaga **skalabilitas** sistem.

#### 1.3.2. Lapisan Kedaulatan Pasien (Self-Sovereign Identity - SSI)

Lapisan ini terdiri dari beberapa komponen, yakni sebagai berikut:

- 1) Saat registrasi, setiap pasien dibuatkan **pasangan kunci kriptografi ECDSA** (kunci publik & privat).
- 2) Kunci privat disimpan aman di perangkat pasien dan digunakan sebagai **tanda tangan digital**.
- 3) Pasien menggunakan kunci ini untuk menyetujui setiap permintaan akses data, menciptakan **bukti persetujuan yang tak terbantahkan** secara matematis.

#### 1.3.3. Lapisan Konsensus & Integritas (Blockchain)

Lapisan ini terdiri dari beberapa komponen, yakni sebagai berikut:

- 1) Blockchain **tidak menyimpan data medis**, melainkan mencatat **sidik jari digital (hash)** dari transaksi penting.
- 2) Hash disimpan ke dalam **Smart Contract** di jaringan Ethereum (disimulasikan dengan **Hardhat**).
- 3) Hasilnya adalah **jejak audit transparan** yang tidak dapat dimanipulasi, menjamin integritas dan kepercayaan.

### 1.4. Target Pengguna

MVP RekamedChain dirancang untuk melayani dua kelompok pengguna utama:

#### 1.4.1. Pasien

- 1) Mendaftar dan membuat identitas digital (SSI)
- 2) Melihat riwayat medis pribadi
- 3) Mengelola permintaan izin akses dari tenaga medis



- 4) Melihat log audit akses data mereka

#### 1.4.2. Tenaga Medis (Dokter)

- 1) Mencari data pasien melalui portal web
- 2) Mengirim permintaan izin akses
- 3) Melihat riwayat medis pasien (setelah izin diberikan)
- 4) Mencatat rekam medis baru dan mengunggah lampiran

### 1.5. Ringkasan Teknologi yang Digunakan

#### 1.5.1. Frontend (Aplikasi Pasien)

- 1) **Framework:** React Native (Expo)
- 2) **Bahasa:** TypeScript
- 3) **Navigasi:** Expo Router (Tab & Stack Navigation)
- 4) **Styling:** React Native StyleSheet
- 5) **Kriptografi:** Ethers.js (pembuatan tanda tangan digital)

#### 1.5.2 Frontend (Portal Dokter)

- 1) **Framework:** Next.js (App Router)
- 2) **Bahasa:** TypeScript
- 3) **Styling:** Tailwind CSS
- 4) **UI Components:** Shadcn/ui

#### 1.5.3 Backend

- 1) **Bahasa:** Go (Golang)
- 2) **Framework:** net/http (standard library)
- 3) **Routing:** ServeMux (awalnya gorilla/mux)
- 4) **Database Driver:** pgx (PostgreSQL)
- 5) **Blockchain Interaction:** go-ethereum (Geth)
- 6) **Autentikasi:** jwt-go
- 7) **CORS Handling:** rs/cors

#### 1.5.4 Database & Penyimpanan

- 1) **Database Utama:** PostgreSQL (Dockerized)
- 2) **Penyimpanan File:** IPFS (Kubo, Dockerized)



### 1.5.5 Blockchain

- 1) **Jaringan Lokal:** Hardhat (Node Ethereum lokal)
- 2) **Bahasa Smart Contract:** Solidity

### 1.5.6 Infrastruktur & DevOps

- 1) **Containerization:** Docker & Docker Compose
- 2) **CI/CD:** GitHub Actions
- 3) **Development Tunneling:** Ngrok

## 1.6. Kesimpulan Singkat

Proyek RekamedChain berhasil mengembangkan sebuah Minimum Viable Product (MVP) yang membuktikan kelayakan konsep rekam medis berbasis blockchain dan SSI.

Melalui arsitektur tiga lapis yang solid, aplikasi ini menunjukkan bahwa:

- 1) Teknologi blockchain dapat menjamin integritas data medis,
- 2) SSI mampu memberi pasien kendali penuh atas data mereka, dan
- 3) Penyimpanan terdesentralisasi dapat menjaga keamanan sekaligus efisiensi sistem.

Walau masih tahap awal, fondasi teknis dan arsitektur yang dibangun sudah sangat kokoh untuk dikembangkan menjadi pilot project skala penuh di masa mendatang.

## 2. Dokumentasi Fitur dan Implementasi

### 2.1. Autentikasi & Manajemen Peran

Fondasi sistem yang mengatur identitas, akses, dan hak pengguna.

Tabel 1 Rincian Fitur Autentikasi dan Manajemen Peran

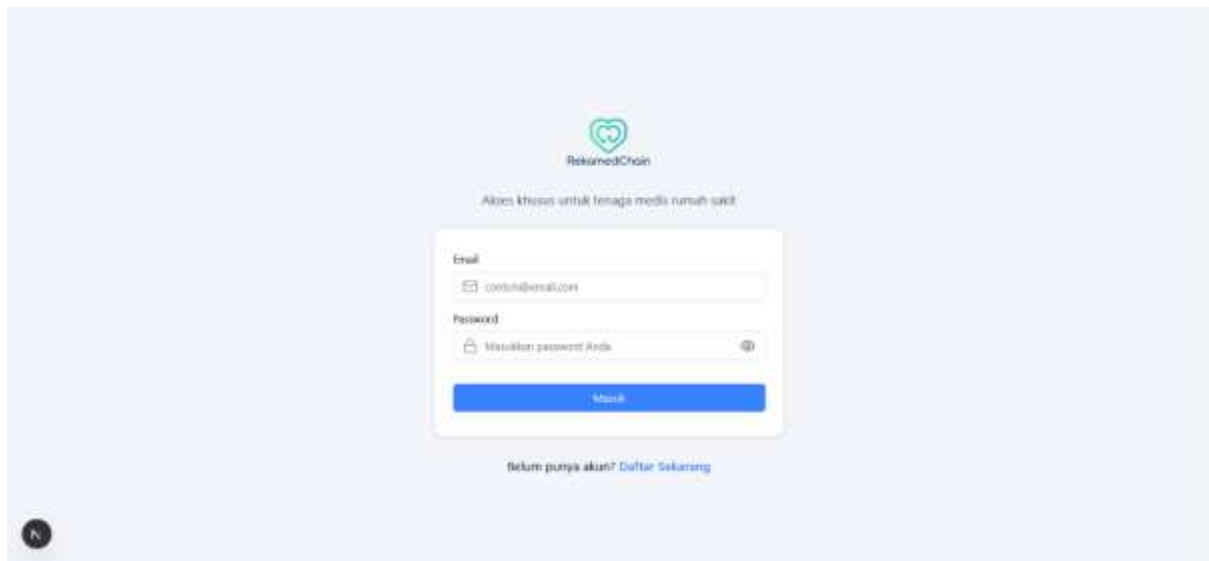
Fitur	Deskripsi	Status	Antarmuka Terkait
<b>Registrasi Pengguna</b>	Pengguna dapat membuat akun baru. Sistem secara otomatis membuat identitas digital dan membedakan peran antara patient (via aplikasi mobile) dan doctor (via portal web)	Sudah Diimplementasikan	Halaman Daftar Akun Baru di Aplikasi Mobile dan Portal Web
<b>Login Pengguna</b>	Pengguna dapat masuk ke sistem menggunakan kredensial (email dan password). Sistem	Sudah Diimplementasikan	Halaman Login di Aplikasi Mobile dan Portal Web

memanfaatkan JSON Web Token (JWT) untuk manajemen sesi yang aman.

<b>Kontrol Akses Berbasis Peran</b>	Sistem membedakan hak akses. Fitur kritis seperti pencatatan rekam medis hanya dapat diakses oleh pengguna dengan peran doctor.	Sudah Diimplementasikan	Terimplementasi di backend (middleware) dan logika UI.
-------------------------------------	---	-------------------------	--

## Screenshot:

Gambar 1 Halaman Registrasi Portal Web



Gambar 2 Halaman Login Portal Web

## 2.2. Alur Rekam Medis & Data

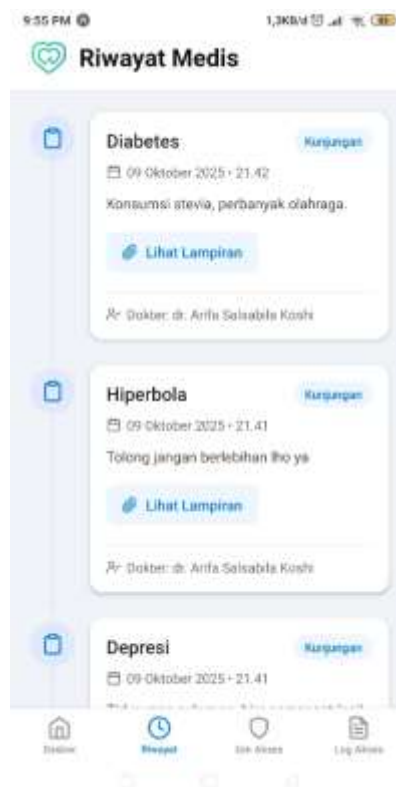
Fitur inti yang berkaitan dengan pengelolaan dan visualisasi data kesehatan.

Tabel 2 Rincian Fitur Rekam Medis dan Data

Fitur	Deskripsi	Status	Antarmuka Terkait
<b>Pembuatan Rekam Medis</b>	Tenaga medis ( <i>doctor</i> ) dapat membuat catatan rekam medis baru untuk pasien, mencakup diagnosis dan observasi.	Sudah Diimplementasikan	Form Tambah Rekam Medis Baru di Portal Web
<b>Penyimpanan Lampiran (IPFS)</b>	File lampiran (misal hasil lab) diunggah ke jaringan IPFS. Alamat unik file (CID) disimpan di database.	Sudah Diimplementasikan	Input file di form Tambah Rekam Medis dan Link Lihat Lampiran
<b>Visualisasi Riwayat Medis</b>	Pasien dan Dokter (dengan izin) dapat melihat riwayat rekam medis secara kronologis.	Sudah Diimplementasikan	Tab Riwayat Medis di Aplikasi Mobile dan halaman Detail Pasien di Portal Web

Screenshot:

Gambar 3 Tambah Rekam Medis Baru



Gambar 4 Riwayat Medis Pasien

### 2.3. Kedaulatan Pasien & Keamanan Data

Fitur inovatif yang menjadi pilar utama RekamedChain.

Tabel 3 Rincian Fitur Autentikasi dan Manajemen Peran

Fitur	Deskripsi	Status	Antarmuka Terkait
<b>Fondasi Self-Sovereign Identity</b>	Saat registrasi, pasien	Sudah Diimplemetasikan	Proses otomatis di backend,

(SSI)	dibuatkan pasangan kunci publik & privat (ECDSA). Kunci privat disimpan aman di perangkat pasien.		penyimpanan kunci di AsyncStorage
Manajemen Izin Akses (Consent)	Pasien dapat menyetujui/menolak permintaan akses data dari dokter. Validasi menggunakan tanda tangan digital pasien.	Sudah Diimplementasikan	Tab Manajemen Izin di Aplikasi Mobile
Enkripsi Data At-Rest (AES)	Data rekam medis dienkripsi AES-256 sebelum disimpan di database untuk menjaga kerahasiaan.	Sudah Diimplementasikan	Proses transparan di backend

## Screenshot:



Gambar 5 Manajemen Izin Akses

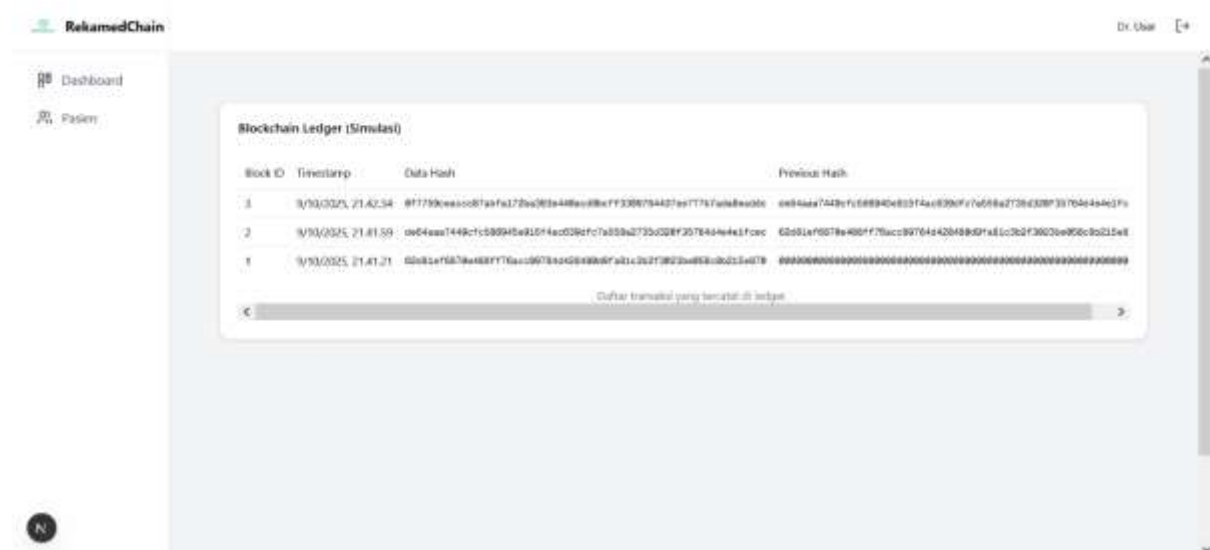
## 2.4. Transparansi & Jejak Audit

Menjamin aktivitas sistem dapat diverifikasi dengan transparan.

Tabel 4 Rincian Fitur Transparansi & Jejak Audit

Fitur	Deskripsi	Status	Antarmuka Terkait
<b>Pencatatan ke Blockchain (Simulasi)</b>	Setiap pembuatan rekam medis mencatat <i>hash</i> ke Smart Contract di jaringan Ethereum lokal (Hardhat).	Sudah Diimplementasikan	Halaman Ledger di Portal Web
<b>Log Akses Pasien</b>	Pasien dapat melihat aktivitas akses terhadap data medis mereka (permintaan, persetujuan, pencatatan baru).	Sudah Diimplementasikan	Tab Log Akses di Aplikasi Mobile

### Screenshot:



Gambar 6 Simulasi Blockchain (Lihat Ledger)

## 2.5. Fitur yang Direncanakan (Menuju Skala Penuh)

Langkah-langkah strategis untuk melengkapi MVP dan mempersiapkan pilot project.

Tabel 5 Fitur yang Direncanakan

Fitur	Deskripsi	Status	Antarmuka Terkait
<b>Bukti Konsep</b>	Implementasi	Direncanakan	Tombol Tarik Data

<b>integrasi</b>	mock API untuk simulasi integrasi eksternal (SATUSEHAT).		di halaman Detail Pasien
<b>Login Tanpa Password (SSI Penuh)</b>	Login sepenuhnya berbasis tanda tangan pasien.	Direncanakan	Alur login baru di aplikasi mobile dan portal web.
<b>Notifikasi Real-Time</b>	Sistem push notification untuk permintaan izin akses baru.	Direncanakan	Fitur notifikasi di aplikasi mobile.

## 2.6. Kesimpulan

Seluruh fitur utama yang direncanakan untuk MVP telah berhasil diimplementasikan dan terintegrasi dengan baik antara komponen frontend (React Native & Next.js), backend (Golang), serta blockchain (Hardhat + Solidity).

Dokumentasi ini menjadi dasar bagi pengembangan tahap berikutnya menuju versi pilot project skala penuh yang siap diujicobakan pada lingkungan kesehatan nyata.

## 3. Arsitektur Sistem: RekamedChain

### 3.1. Penjelasan Singkat Alur Sistem

Arsitektur RekamedChain dirancang sebagai sistem multi-lapis yang memisahkan antarmuka pengguna, logika bisnis, dan lapisan data.

Alur sistem secara umum adalah sebagai berikut:

#### 3.1.1. Interaksi Pengguna

Pengguna (Pasien atau Dokter) berinteraksi melalui aplikasi antarmuka sesuai perannya:

- 1) **Aplikasi Mobile (React Native)** untuk Pasien
- 2) **Portal Web (Next.js)** untuk Dokter

#### 3.1.2. Permintaan ke Backend

Setiap aksi pengguna diterjemahkan menjadi permintaan HTTPS ke **Backend API Gateway (Go)**.

#### 3.1.3. Proses di Backend

Backend API memproses permintaan: validasi, otentikasi (JWT), otorisasi (role-based), enkripsi/dekripsi data, dan interaksi dengan database.



### 3.1.4. Interaksi dengan Lapisan Data

Backend berkomunikasi dengan tiga komponen penyimpanan:

- 1) **PostgreSQL**: menyimpan data terstruktur seperti profil dan metadata rekam medis
- 2) **IPFS**: menyimpan file biner (lampiran medis)
- 3) **Blockchain (Hardhat)**: mencatat hash transaksi sebagai jejak audit yang kekal (immutable)

### 3.1.5. Respon ke Frontend

Hasil pemrosesan dikirim kembali ke frontend untuk ditampilkan kepada pengguna.

## 3.2. Daftar Teknologi dan Framework

Berikut rincian **tech stack** yang digunakan dalam proyek RekamedChain:

### 3.2.1. Frontend (Aplikasi Pasien)

- 1) **Framework**: React Native (Expo)
- 2) **Bahasa**: TypeScript
- 3) **Navigasi**: Expo Router
- 4) **Kriptografi**: Ethers.js
- 5) **Penyimpanan Lokal**: AsyncStorage

### 3.2.2. Frontend (Portal Dokter)

- 1) **Framework**: Next.js (App Router)
- 2) **Bahasa**: TypeScript
- 3) **Styling**: Tailwind CSS
- 4) **Komponen UI**: Shadcn/ui

### 3.2.3. Backend

- 1) **Bahasa**: Go (Golang)
- 2) **Framework**: net/http (Standard Library)
- 3) **Dependensi Kunci**:
  - (1) pgx → PostgreSQL
  - (2) go-ethereum → Interaksi Blockchain
  - (3) jwt-go → Autentikasi
  - (4) rs/cors → CORS Handling

### 3.2.4. Database & Penyimpanan

- 1) **Database Utama:** PostgreSQL
- 2) **Penyimpanan File:** InterPlanetary File System (IPFS / Kubo)

### 3.2.5. Blockchain

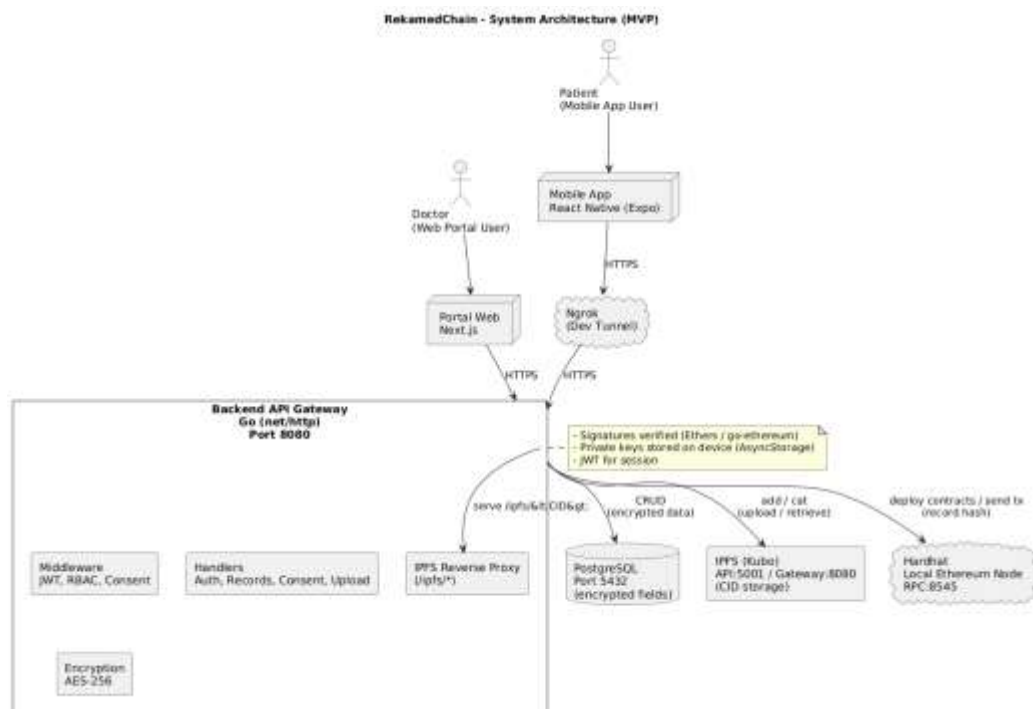
- 1) **Jaringan Development:** Hardhat (Node Ethereum lokal)
- 2) **Bahasa Smart Contract:** Solidity

### 3.2.6. Infrastruktur & DevOps

- 1) **Containerization:** Docker & Docker Compose
- 2) **Tunneling (Development):** Ngrok

### 3.3. Diagram Arsitektur Sistem

Diagram berikut mengilustrasikan komponen-komponen utama dan alur interaksi sistem:



Gambar 7 Diagram Arsitektur Sistem

### 3.4. Penjelasan Hubungan Antar Komponen

#### 3.4.1. Frontend → Backend

Seluruh interaksi (login, unggah, permintaan data) dari frontend diarahkan ke **Backend API Gateway**.

Untuk versi pengembangan mobile, koneksi dijumpatani melalui **Ngrok**.

### 3.4.2. Backend → PostgreSQL

Menggunakan **driver pgx** untuk operasi CRUD terhadap data terstruktur: profil pengguna, metadata rekam medis, status izin, dsb.

### 3.4.3. Backend → IPFS

- 1) **Upload:** Dokter mengunggah file → backend kirim ke IPFS API (port 5001) → dapat CID.
- 2) **Download/View:** Permintaan ke /ipfs/<CID> diteruskan via reverse proxy ke IPFS Gateway (port 8080 internal).

### 3.4.4. Backend → Hardhat

Saat server dijalankan:

- 1) Backend menginisialisasi koneksi ke **node Hardhat (port 8545)**
- 2) Deploy **smart contract Ledger.sol**

Setiap pembuatan rekam medis baru → kirim transaksi ke blockchain → catat hash data.

## 3.5. Ringkasan Keamanan dan Otentikasi

Keamanan merupakan pilar utama sistem RekamedChain.

### 3.5.1. Autentikasi Sesi

- 1) Menggunakan **JWT (JSON Web Token)**.
- 2) Token dikirim di header Authorization untuk setiap request yang dilindungi.

### 3.5.2. Otorisasi Berbasis Peran (RBAC)

- 1) Middleware backend memeriksa role dalam JWT (patient / doctor).
- 2) Endpoint tertentu seperti POST /records hanya dapat diakses oleh dokter.

### 3.5.3. Self-Sovereign Identity (SSI)

Persetujuan akses data (consent) divalidasi menggunakan **tanda tangan digital (ECDSA)**.

Pasien menandatangani permintaan dengan **private\_key** di perangkat; backend memverifikasi dengan **public\_key** di database.

### 3.5.4. Enkripsi Data At-Rest

Data medis sensitif (diagnosis, notes) dienkripsi menggunakan **AES-256** sebelum disimpan di PostgreSQL.

### 3.5.5. Integritas Data

Setiap penambahan rekam medis menghasilkan hash yang dicatat di **blockchain lokal (Hardhat)**.

Hal ini menjamin **audit trail** yang immutable dan transparan.

#### 4. Database Design

##### 4.1. Penjelasan Struktur Database yang Digunakan

Struktur basis data **RekamedChain** dirancang untuk mendukung sistem **rekam medis terdesentralisasi** dengan fokus pada keamanan, privasi, dan keterlacakan data pasien.

Pada tahap **MVP (Minimum Viable Product)**, desain difokuskan pada fungsi utama yaitu:

- 1) Pendaftaran pengguna (dokter & pasien)
- 2) Pembuatan dan penyimpanan rekam medis
- 3) Manajemen izin akses (consent)
- 4) Pencatatan hash data pada *blockchain ledger* internal

Ke depan, struktur akan dikembangkan untuk mendukung lebih banyak peran (admin, perawat), tipe rekam medis (laboratorium, resep, radiologi), dan integrasi fasilitas kesehatan (*faskes*).

##### 4.2. Daftar Tabel dan Atribut

###### 4.2.1. Tabel Users

Tabel 6 Tabel Users

Kolom	Tipe Data	Keterangan
<b>id</b>	Uuid (PK, uuid_generate_v4()) default	ID unik pengguna
<b>name</b>	varchar	Nama pengguna
<b>Email</b>	Varchar (unique, not null)	Alamat email
<b>Hashed_password</b>	Text	Password yang di-hash
<b>Role</b>	Roletype	Peran pengguna (patient, doctor)
<b>Phone</b>	Varchar	Nomor telepon
<b>Nip</b>	Varchar	Nomor induk pegawai (jika tenaga medis)
<b>Specialization</b>	Varchar	Spesialisasi medis

<b>Public_key</b>	Text	Kunci publik untuk enkripsi data
<b>Private_key_encrypted</b>	text	Kunci privat terenkripsi
<b>Created_at</b>	timestampz	Waktu pembuatan data
<b>Updated_at</b>	Timestampz	Waktu pembaruan data

#### 4.2.2. Tabel MedicalRecords

Tabel 7 Tabel MedicalRecords

Kolom	Tipe Data	Keterangan
<b>id</b>	Uuid (PK)	ID unik rekam medis
<b>Patient_id</b>	Uuid (FK -> Users.id)	Pemilik data (pasien)
<b>Doctor_name</b>	Varchar	Nama dokter pencatat
<b>Diagnosis</b>	Text	Diagnosis dokter
<b>Notes</b>	Text	Catatan tambahan
<b>Attachment_cid</b>	Varchar	Pointer ke file terenkripsi di IPFS/server
<b>Created_at</b>	Timestampz	Waktu pembuatan catatan

#### 4.2.3. Tabel ConsentRequests

Tabel 8 Tabel ConsentRequests

Kolom	Tipe Data	Keterangan
<b>id</b>	Uuid (PK)	ID unik permintaan
<b>Doctor_id</b>	Uuid (FK -> Users.id)	Dokter yang meminta izin
<b>Patient_id</b>	Uuid (FK -> Users.id)	Pasien pemilik data
<b>Status</b>	consentStatus	Status permintaan (pending, approved,

		rejected)
<b>Created_at</b>	Timestamptz	Waktu dibuat
<b>Updated_at</b>	Timestamptz	Waktu diperbarui

#### 4.2.4. Tabel BlockchainLedger

Tabel 9 Tabel BlockchainLedger

Kolom	Tipe Data	Keterangan
<b>Block_id</b>	Int (PK, increment)	Nomor blok
<b>Record_id</b>	Uuid (FK -> MedicalRecords.id)	ID rekam medis terkait
<b>Data_hash</b>	Varchar	Hash dari data rekam medis
<b>Previous_hash</b>	Varchar	Hash dari blok sebelumnya
<b>Created_at</b>	Timestamptz	Waktu pencatatan

### 4.3. Diagram ERD



Gambar 8 ERD untuk MVP saat ini

### 4.4. Penjelasan Relasi Antar Tabel

Struktur database RekamedChain MVP terdiri dari empat tabel utama yang saling berhubungan untuk mendukung alur autentikasi pengguna, pengelolaan rekam medis, kontrol izin akses, dan pencatatan hash ke blockchain.

#### 4.4.1. Relasi Utama

##### 1) **Users** ↔ **MedicalRecords**

(1) Relasi: *One-to-Many*

(2) Satu pengguna dengan peran **pasien** dapat memiliki banyak entri rekam medis.

(3) Atribut relasi: **MedicalRecords.patient\_id** → **Users.id**

##### 2) **Users** ↔ **ConsentRequests**

(1) Relasi: *Many-to-Many* (melalui dua foreign key)



(2) Dokter dapat mengajukan beberapa permintaan izin ke pasien yang berbeda, dan pasien juga dapat menerima beberapa permintaan izin dari berbagai dokter.

(3) Atribut relasi:

- ConsentRequests.doctor\_id → Users.id
- ConsentRequests.patient\_id → Users.id

### 3) **MedicalRecords ↔ BlockchainLedger**

(1) Relasi: *One-to-One*

(2) Setiap entri rekam medis memiliki satu catatan terkait di tabel BlockchainLedger yang menyimpan hash dari data medis untuk menjamin integritas dan transparansi.

(3) Atribut relasi: BlockchainLedger.record\_id → MedicalRecords.id

### 4) **Users ↔ BlockchainLedger (secara tidak langsung)**

(1) Relasi: *Indirect via MedicalRecords*

(2) Pengguna (pasien atau dokter) berkontribusi terhadap catatan yang akhirnya di-hash dan dicatat pada blockchain, memberikan jejak audit atas aktivitas data medis.

## 4.5. Alasan Perancangan

Desain database RekamedChain MVP mengikuti prinsip **kesederhanaan, skalabilitas, dan keamanan**. Beberapa pertimbangan perancangan antara lain:

### 1) **Pemecahan Peran dalam Tabel Users**

- (1) Alih-alih membuat tabel terpisah untuk dokter dan pasien, MVP hanya menggunakan satu tabel Users dengan atribut role untuk membedakan peran.
- (2) Pendekatan ini mempercepat pengembangan awal dan memudahkan autentikasi terpusat (single user table).

### 2) **Tabel MedicalRecords yang Modular**

- (1) Tabel ini menyimpan data inti dari setiap rekam medis, seperti diagnosis, catatan tambahan, dan CID file di IPFS.
- (2) Penyimpanan attachment\_cid memungkinkan integrasi dengan sistem terdistribusi tanpa membebani database utama.

### 3) **ConsentRequests sebagai Pengontrol Akses Data**

- (1) Dengan model izin berbasis permintaan, pasien memiliki kontrol penuh terhadap siapa yang dapat mengakses data medis mereka.
- (2) Status pending, approved, dan rejected mendukung alur otorisasi yang eksplisit.

#### 4) **Integrasi Blockchain Melalui BlockchainLedger**

- (1) Pencatatan hash dari data medis di tabel BlockchainLedger berfungsi sebagai lapisan audit dan verifikasi integritas.
- (2) Desain ini memungkinkan validasi historis tanpa mengungkapkan data medis asli.

#### 5) **Kesiapan untuk Skalabilitas**

Struktur ini dirancang agar mudah diperluas di masa depan. Misalnya:

- (1) Penambahan tabel Patients dan HealthProfessionals untuk menyimpan data lebih rinci.
- (2) Penambahan enumerasi RecordType untuk mendukung berbagai jenis rekam medis (konsultasi, resep, hasil lab, dll).

#### 6) **Keseimbangan antara MVP dan Visi Produk Akhir**

- (1) MVP difokuskan pada fungsi inti (akses data medis aman dan transparan), namun tetap kompatibel dengan desain penuh yang telah direncanakan.
- (2) Hal ini memungkinkan transisi bertahap tanpa perlu migrasi database besar di kemudian hari.

### 5. Use Case Diagram dan Deskripsi: RekamedChain

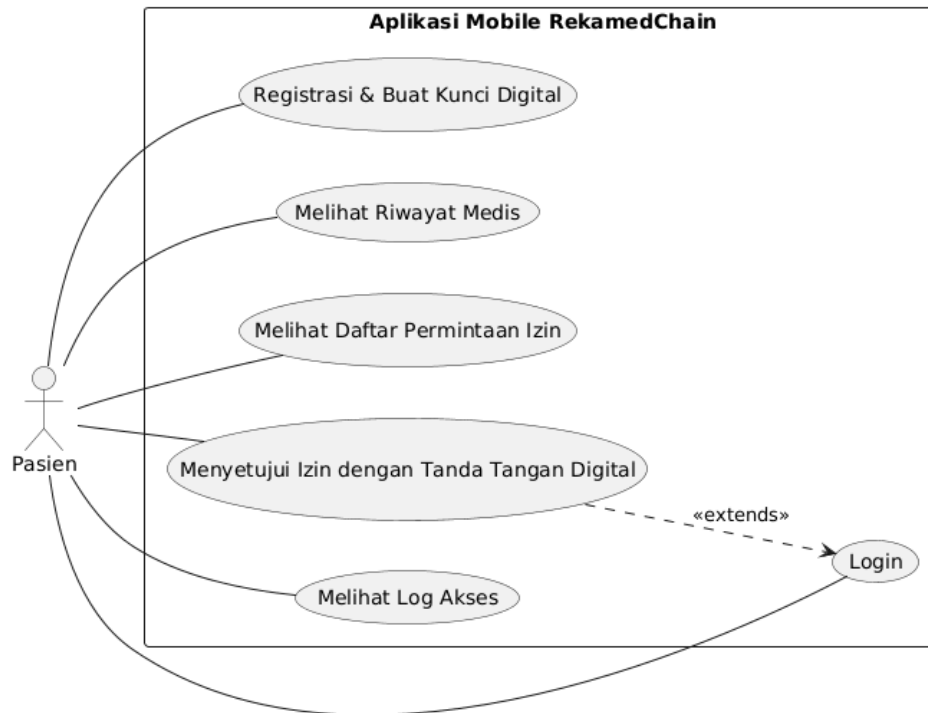
#### 5.1. Daftar Aktor

Sistem RekamedChain memiliki dua aktor utama yang berinteraksi langsung dengan aplikasi:

- 1) **Pasien:** Pengguna utama yang merupakan pemilik data medis. Pasien memiliki kedaulatan penuh untuk mengelola dan memberikan persetujuan akses terhadap data kesehatannya melalui aplikasi mobile.
- 2) **Tenaga Medis (Dokter):** Pengguna profesional yang memiliki wewenang untuk mencatat data medis dan meminta akses ke riwayat medis pasien setelah mendapatkan persetujuan. Dokter berinteraksi dengan sistem melalui portal web.

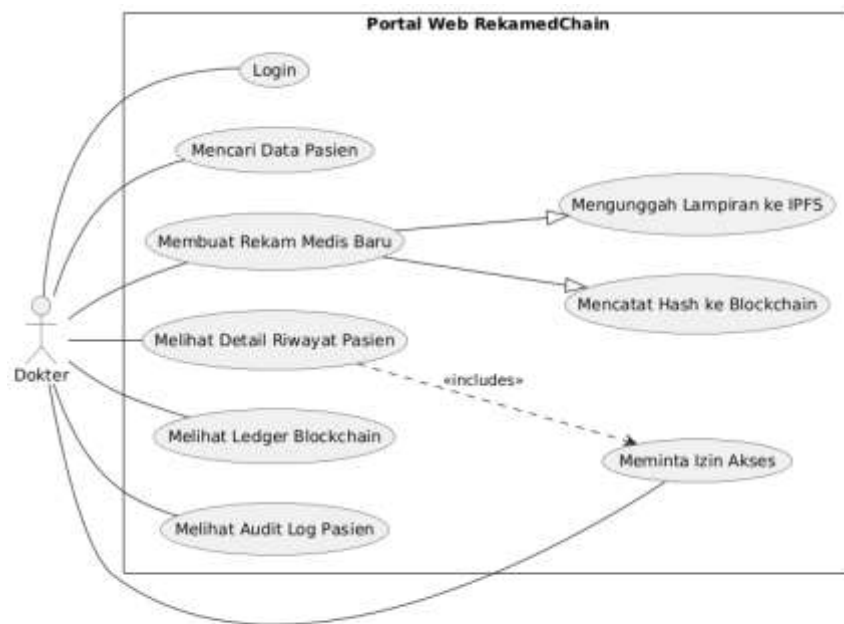
## 5.2. Diagram Use Case

### 5.2.1. Diagram Use Case Pasien



Gambar 9 Use Case Diagram Pasien

### 5.2.2. Diagram Use Case Tenaga Medis (Dokter)



Gambar 10 Use Case Diagram Dokter

### 5.3. Daftar dan Deskripsi Use Case

#### 5.3.1. Aktor: Pasien

Tabel 10 Deskripsi Use Case Pasien

Use Case	Deskripsi (Fungsi dan Manfaat)	Status	Antarmuka Terkait
<b>Registrasi dan Pembuatan Kunci Digital</b>	Pasien membuat akun baru melalui aplikasi mobile. Sistem secara otomatis menghasilkan pasangan kunci kriptografi (publik & privat) yang menjadi dasar identitas digital berdaulat (SSI) pasien. Manfaat: Memberikan pasien alat (kunci privat) untuk mengontrol datanya secara mandiri.	Sudah Diimplementasikan	-

<b>Login</b>		Pasien masuk ke aplikasi menggunakan email dan password untuk mengakses fitur-fitur yang dipersonalisasi	Sudah diimplementasikan	-
<b>Melihat Medis</b>	<b>Riwayat</b>	Pasien dapat melihat seluruh riwayat medisnya dalam format linimasa yang kronologis dan mudah dibaca, termasuk detail diagnosis, catatan dokter, dan lampiran. Manfaat: Memberikan transparansi dan akses mudah ke data kesehatan pribadi.	Sudah diimplementasikan	-
<b>Mengelola Akses</b>	<b>Izin</b>	Pasien dapat melihat daftar permintaan akses dari tenaga medis, lengkap dengan detail siapa yang meminta dan untuk tujuan apa. Pasien memiliki kontrol penuh untuk menyetujui atau menolak permintaan tersebut. Manfaat: Ini adalah inti dari kedaulatan pasien.	Sudah diimplementasikan	-
<b>Menyetujui Tanda Digital</b>	<b>dnegan Tangan</b>	Saat menyetujui permintaan, aplikasi menggunakan kunci privat pasien untuk membuat tanda tangan digital yang unik. Manfaat: Memberikan persetujuan yang terbukti secara kriptografis, aman, dan tidak dapat disangkal.	Sudah diimplementasikan	Tombol “Setujui” di halaman Manajemen Izin

<b>Melihat Log Akses</b>	Pasien dapat melihat jejak audit transparan dari semua aktivitas yang terjadi pada datanya, seperti kapan izin diminta, disetujui, dan kapan data baru ditambahkan. Manfaat: Meningkatkan transparansi dan kepercayaan.	Sudah diimplementasikan	-
--------------------------	---	-------------------------	---

### 5.3.2 Aktor: Tenaga Medis (Dokter)

Tabel 11 Deskripsi Use Case Dokter

Use Case	Deskripsi (Fungsi dan Manfaat)	Status	Antarmuka Terkait
<b>Login</b>	Dokter masuk ke portal web untuk mengakses fitur-fitur profesional.	Sudah Diimplementasikan	-
<b>Mencari Pasien</b>	Dokter dapat mencari data pasien yang terdaftar di sistem berdasarkan nama atau email untuk memulai proses interaksi medis.	Sudah diimplementasikan	Halaman "Pencarian Pasien" di portal web
<b>Meminta Izin Akses</b>	Dari halaman detail pasien, dokter dapat mengirimkan permintaan untuk mengakses riwayat medis pasien. Permintaan ini akan dikirim ke aplikasi pasien.	Sudah diimplementasikan	Tombol "Minta Izin Akses" di halaman Detail Pasien
<b>Melihat Detail Riwayat Pasien</b>	Setelah mendapatkan izin, dokter dapat melihat riwayat medis lengkap pasien	Sudah diimplementasikan	-

dalam format linimasa terpadu. Manfaat: Membantu dokter membuat diagnosis yang lebih akurat dan cepat.

#### Membuat Rekam Medis Baru

Dokter dapat mencatat data medis baru (diagnosis, catatan, lampiran). Data sensitif akan dienkripsi, lampiran disimpan ke IPFS, dan hash-nya dicatat ke blockchain. Manfaat: Memastikan data baru aman, terverifikasi, dan terintegrasi.

Sudah diimplementasikan

-

#### Melihat Blockchain

##### Ledger

Dokter dapat melihat “buku besar” blockchain yang berisi daftar hash dari semua rekam medis yang pernah dibuat, sebagai bukti integritas data.

Sudah diimplementasikan

Halaman “Lihat Ledger” di portal web

#### Melihat Audit Log Pasien

Setelah mendapat izin, dokter dapat melihat jejak audit aktivitas pada data seorang pasien.

Sudah diimplementasikan

Halaman “Audit Log” di portal web

#### Integrasi SATUSEHAT

Dokter dapat menyimulasikan penarikan data pasien dari sistem eksternal seperti SATUSEHAT. Manfaat: Membuktikan kapabilitas interoperabilitas sistem RekamedChain.

Direncanakan

Tombol “Tarik Data” di halaman Detail Pasien



## 6. Desain UI/UX: RekamedChain

### 6.1. Makna Logo



Gambar 11 Logo RekamedChain

Logo RekamedChain adalah fondasi dari identitas visual dan filosofi proyek. Setiap elemen dirancang untuk menceritakan sebuah narasi:

- 1) **Bentuk Hati:** Melambangkan pilar utama yaitu kesehatan, kepedulian, dan fokus pada kemanusiaan (pasien).
- 2) **Jalinan di Dalam Hati:** Merepresentasikan tiga konsep inti:
  - (1) **Rantai (Chain):** Simbol dari keamanan dan integritas yang dibawa oleh teknologi Blockchain.
  - (2) **Integrasi:** Visualisasi dari solusi atas masalah data yang terfragmentasi, di mana RekamedChain menyatukan riwayat medis dari berbagai faskes.
  - (3) **Hubungan Kepercayaan:** Menggambarkan hubungan berbasis persetujuan (consent) antara pasien dan tenaga medis.
- 3) **Garis Kontinu:** Melambangkan kesinambungan perawatan (continuity of care), di mana data pasien mengalir utuh dan tanpa putus sepanjang hidup mereka.
- 4) **Gradasi Warna (Hijau ke Biru):** Menunjukkan sinergi yang harmonis antara dunia kesehatan (hijau) dan teknologi/kepercayaan (biru).

Secara keseluruhan, logo ini bermakna: “Sebuah sistem yang berpusat pada kepedulian terhadap pasien (hati), dengan mengintegrasikan teknologi blockchain yang aman (jalinan rantai) untuk menciptakan alur riwayat medis yang berkelanjutan (garis kontinu).”

### 6.2. Prinsip Desain dan Gaya Visual

Desain RekamedChain dibangun di atas tiga pilar utama: **Kepercayaan, Kejelasan, dan Kedaulatan.**

## 1) Palet Warna Utama

- (1) **Biru Utama (#007AFF)**: Kepercayaan, teknologi, stabilitas. Digunakan untuk aksi primer (tombol utama, tautan, ikon aktif).
- (2) **Hijau Sekunder (#34D399)**: Kesehatan, persetujuan, status “Aktif”. Digunakan untuk badge status positif.
- (3) **Merah Aksen (#EF4444)**: Peringatan, penolakan, aksi “Cabut”. Digunakan terbatas untuk menarik perhatian tindakan penting.
- (4) **Netral**: Abu-abu (#f0f4f8 untuk latar) & putih (#ffffff untuk kartu) untuk tampilan bersih, minimalis, fokus konten.

## 2) Tipografi

Menggunakan font sans-serif modern & mudah dibaca (System UI di mobile, Inter di web).

## 3) Gaya Ikon

Library **Feather Icons**: outline, modern, minimalis, selaras dengan prinsip desain fungsional.

### 6.3. Alur Navigasi (Flow)

#### **Alur Pasien (Aplikasi Mobile):**

- 1) Buka Aplikasi → diarahkan ke Login
- 2) Login/Register → Pengguna membuat akun (menyimpan private key)
- 3) Dasbor Utama (Tab Navigator: Dasbor, Riwayat, Izin Akses, Log Akses)
- 4) Logout → Tombol di halaman profil (fitur direncanakan)

#### **Alur Tenaga Medis (Portal Web):**

- 1) Buka Portal → diarahkan ke Login
- 2) Login → Dokter masuk ke sistem
- 3) Navigasi via Navbar → Pencarian Pasien, Ledger Blockchain, Logout
- 4) Alur Inti: Pencarian Pasien → Detail Pasien → Meminta Izin / Melihat Audit Log / Menambah Rekam Medis Baru

### 6.4. Deskripsi Halaman Utama dan Komponennya

#### 6.4.1. Aplikasi Mobile (Pasien)

- 1) **Halaman Login & Registrasi**: Layout berbasis kartu, logo, input dengan ikon, show/hide password, modal animasi Lottie untuk feedback.

- 2) **Halaman Dasbor:** Header profil, kartu notifikasi “Permintaan Akses Baru”, kartu “Ringkasan Kesehatan”, kartu aksi navigasi.
- 3) **Halaman Riwayat, Izin, Log Akses:** Layout linimasa atau SectionList, data kronologis, badge warna untuk status.

#### 6.4.2. Portal Web (Dokter)

- 1) **Layout Utama:** Navbar persisten untuk navigasi konsisten
- 2) **Halaman Pencarian & Detail:** Layout kartu, informasi kondisional berdasarkan status izin
- 3) **Komponen:** Menggunakan **shadcn/ui** untuk konsistensi dan aksesibilitas

#### 6.5. Bagaimana UI Mendukung UX

Desain UI RekamedChain mendukung pengalaman pengguna (UX) dengan menerapkan prinsip heuristik Jakob Nielsen:

- 1) **Visibility of System Status:** Feedback loading, error jelas, modal animasi Lottie untuk status sukses/gagal.
- 2) **User Control and Freedom:** Show/hide password, tombol kembali, tombol “Cabut Izin” → kendali penuh pengguna.
- 3) **Consistency and Standards:** Tab navigator di mobile & navbar di web konsisten, ikonografi standar → mengurangi beban kognitif.
- 4) **Aesthetic and Minimalist Design:** Layout bersih, fokus konten, penggunaan whitespace & kartu memudahkan pencarian informasi.
- 5) **Branding & Logo:** Memperkuat identitas aplikasi dan meningkatkan kepercayaan pengguna.

### 7. Rencana Pengembangan: RekamedChain

#### 7.1. Evaluasi Versi MVP

Versi MVP saat ini telah membuktikan kelayakan teknis dari konsep inti proyek. Alur kerja utama, dari pembuatan identitas digital pasien hingga pencatatan data medis yang diamankan oleh simulasi blockchain, telah berhasil diimplementasikan.

##### 7.1.1. Kelebihan & Pencapaian Utama

- 1) **Validasi Konsep:** Arsitektur tiga lapis: penyimpanan off-chain (PostgreSQL & IPFS), lapisan kedaulatan pasien (SSI), dan lapisan integritas (blockchain ledger), telah terbukti dapat diimplementasikan secara sinergis.
- 2) **Prototipe Fungsional:** Dua antarmuka utama (Aplikasi Pasien Mobile & Portal Tenaga Medis Web) dibangun dengan alur fitur esensial yang berfungsi penuh.

### 3) Alur Pengguna Lengkap:

Registrasi Pasien (SSI) → Login → Permintaan Izin oleh Dokter → Persetujuan Izin oleh Pasien (tanda tangan digital) → Dokter Melihat & Menambah Rekam Medis → Data Baru Tercatat di Blockchain Ledger.

### 4) **Fondasi Teknologi Solid:** Infrastruktur berbasis Docker → lingkungan konsisten, portabel, mudah dikelola.

#### 7.1.2. Kendala & Area Peningkatan

- 1) **UX Belum Optimal:** Beberapa antarmuka mobile masih menggunakan data dummy (nama pasien & detail dokter) karena keterbatasan endpoint API.
- 2) **Implementasi SSI Belum Penuh:** Login masih menggunakan email/password → sistem passwordless belum diterapkan.
- 3) **Proses Manual:** Migrasi database & pembuatan Go binding smart contract masih manual → rawan human error.

#### 7.2. Daftar Fitur Lanjutan yang Direncanakan

Untuk evolusi dari MVP ke produk skala penuh:

- 1) **Arsitektur Backend Modular:** Refactoring total backend ke struktur package (api, storage, domain, utils, dll.).
- 2) **Manajemen Profil Pengguna:** Endpoint untuk mengambil detail profil pengguna sendiri & pengguna lain berdasarkan ID.
- 3) **Login Tanpa Password (Passwordless):** Otentikasi berbasis SSI dengan tanda tangan digital “pesan tantangan”.
- 4) **Manajemen Izin Penuh:** Fitur menolak & mencabut izin, durasi izin (misal 24 jam vs permanen).
- 5) **Notifikasi Real-time:** Push notification untuk permintaan izin baru di aplikasi mobile.
- 6) **Enkripsi Tingkat Lanjut (CP-ABE):** Implementasi Ciphertext-Policy Attribute-Based Encryption untuk kontrol akses granular.
- 7) **Proof-of-Concept Integrasi Eksternal:** Simulasi tarik data dari API eksternal (misal SATUSEHAT) → uji interoperabilitas.

#### 7.3. Roadmap Pengembangan dan Prioritas

Pengembangan dibagi tiga tahap dengan fokus berbeda.

##### 7.3.1. Tahap 1: Penguatan Fondasi & UX

**Prioritas Utama | Estimasi: 1-3 Hari**

Tujuan: Lunasi utang teknis, hilangkan data dummy, aplikasi terasa lengkap bagi pengguna.

P1: Integrasi Endpoint Mobile dengan Front-end

### 7.3.2. **Tahap 2: Implementasi Kedaulatan Penuh & Interaktivitas** **Prioritas Tinggi | Estimasi: 1 Minggu**

Tujuan: Wujudkan visi SSI penuh & interaktivitas lebih hidup.

**P1:** Implementasi Login Tanpa Password (Passwordless)

**P2:** Implementasi Notifikasi Real-time permintaan izin

**P3:** Sempurnakan Detail Izin → logika durasi & lingkup data

### 7.3.3. **Tahap 3: Keamanan Lanjutan & Visi Jangka Panjang** **Prioritas Menengah | Estimasi: 2-4 Minggu**

Tujuan: Implementasi fitur paling canggih.

**P1:** Riset & Implementasi Awal CP-ABE

**P2:** Proof-of-Concept Integrasi SATUSEHAT

**P3:** Migrasi Smart Contract ke Testnet Publik (misal: Sepolia)

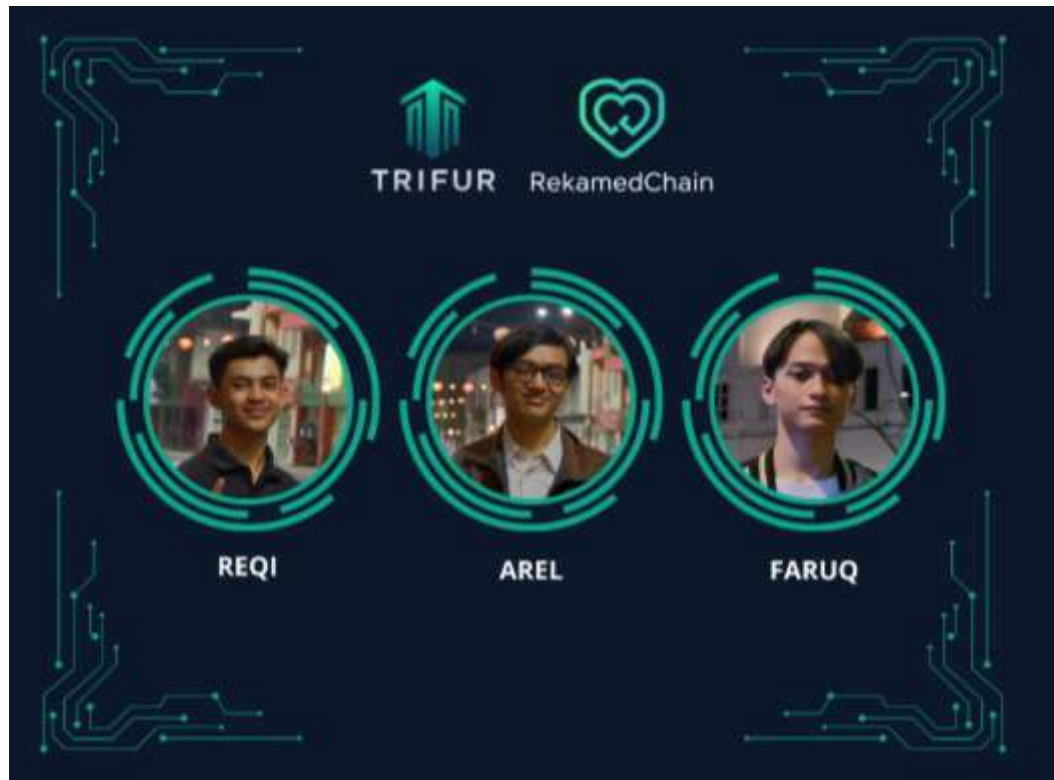
### 7.4. **Potensi Ekspansi**

Setelah MVP skala penuh, potensi ekspansi sangat luas:

- 1) **Integrasi Penuh dengan API Nasional:** SATUSEHAT, BPJS, SIMRS
- 2) **Modul Tambahan:** Peran lain → Apotek (validasi resep), Laboratorium (hasil tes).
- 3) **Migrasi ke Hyperledger Fabric:** Dari Ethereum MVP → Hyperledger enterprise-grade.
- 4) **Implementasi Verifiable Credentials (VCs):** Upgrade model SSI menggunakan W3C VCs, pasien memegang sertifikat digital terpercaya (misal bukti vaksinasi).

## 8. **Kontribusi Tim dan Pembagian Peran: Tim Trifur**

## 8.1. Daftar Anggota Tim



*Gambar 12 Anggota Trifur*

Tim inti Trifur terdiri dari tiga anggota dengan keahlian komplementer yang mencakup spektrum penuh dalam pengembangan produk digital:

- 1) **Farrel Zandra**
- 2) **Reqi Jumentara Hapid**
- 3) **Umar Faruq Robbany**

## 8.2. Peran dan Tanggung Jawab

Setiap anggota memegang peran ganda (hybrid roles) yang memungkinkan tim bergerak lincah dan adaptif, ciri khas tim startup yang efektif.

**Tabel 12 Peran dan Tanggung Jawab**

<b>Nama</b>	<b>Peran Utama</b>	<b>Tanggung Jawab Kunci</b>
<b>Farrel Zandra</b>	Project Manager & Full Stack Developer	Visi produk, perencanaan strategis, arsitektur sistem, pengembangan backend inti, dan manajemen infrastruktur.
<b>Reqi Jumentara Hapid</b>	UI/UX Designer & Backend	Desain antarmuka &

	Developer	pengalaman pengguna (mockup), riset pengguna, pengembangan fitur backend.
<b>Umar Faruq Robbany</b>	Frontend Developer & Quality Assurance	Implementasi desain ke kode frontend (Web & Mobile) dan menyusun strategi pengujian kualitas aplikasi untuk penjaminan mutu.

### 8.3. Deskripsi Kontribusi Selama Pengembangan

#### 8.3.1. Farrel Zandra

Sebagai motor penggerak proyek, Farrel mendefinisikan visi awal dan roadmap MVP. Kontribusi teknis utama:

- 1) Membangun fondasi backend dari nol, termasuk skema database.
- 2) Mengimplementasikan logika autentikasi (JWT).
- 3) Integrasi enkripsi (AES), penyimpanan IPFS, dan menginisiasi interaksi smart contract.
- 4) Bertanggung jawab atas arsitektur DevOps: konfigurasi Docker, CI/CD, dan ngrok → lingkungan pengembangan solid.

#### 8.3.2. Reqi Jumantera Hapid

Sebagai arsitek pengalaman pengguna:

- 1) Merancang desain visual & alur aplikasi profesional (mockup awal proposal).
- 2) Menerjemahkan kebutuhan pasien & dokter menjadi UI intuitif.
- 3) Berkolaborasi di backend, memimpin refactoring untuk modularitas & skalabilitas.
- 4) Membuat Endpoint yang berguna untuk Front-end.
- 5) Melakukan pengujian endpoint yang telah dibuat.

#### 8.3.3. Umar Faruq Robbany

Sebagai eksekutor frontend:

- 1) Menerjemahkan mockup menjadi kode hidup di Web (Next.js) & Mobile (React Native/Expo).
- 2) Implementasi semua halaman, dari login/register hingga dashboard dinamis.
- 3) Memastikan komponen UI/UX berfungsi optimal.



- 4) Menyusun strategi dan skenario pengujian aplikasi untuk penjaminan mutu.

#### 8.4. Tools Kolaborasi

Untuk menjaga alur kerja tetap terorganisir dan efisien, tim Trifur memanfaatkan tools industri standar:

- 1) **Manajemen Kode:** GitHub (version control & CI/CD)
- 2) **Desain & Prototyping:** Lovable (mockup & UI/UX)
- 3) **Manajemen Proyek:** ClickUp (task breakdown & sprint tracking)
- 4) **Dokumentasi & Kolaborasi:** Google Docs (proposal & dokumen internal)
- 5) **Infrastruktur & Development:** Docker, ngrok, DBeaver, Visual Studio Code

#### 8.5. Kesimpulan Kerja Tim

Tim Trifur menunjukkan sinergi luar biasa dengan peran hibrida fleksibel.

- 1) Backend dibantu desainer, frontend didukung manajer proyek → mampu mengatasi tantangan teknis kompleks dengan cepat.
- 2) Kolaborasi erat antara desain, backend, dan frontend → kunci keberhasilan mentransformasi visi ambisius RekamedChain menjadi MVP fungsional siap tahap selanjutnya.