

LAPORAN DOKUMENTASI

I. KONSEP UMUM PROJECT

Proyek ini menerapkan konsep modular blockchain lokal sebagai integrity layer untuk data kredensial akademik. Prinsipnya data yang masuk melalui API dicatat ke dalam rangkaian blok yang saling terhubung melalui hash chain SHA-256, setiap perubahan pada isi blok akan membuat ketidaksesuaian hash dan memutus rantai. Google Sheets diposisikan sebagai media penyimpanan dan tampilan cloud untuk pengujian terhadap manipulas, blockchain lokal bertindak sebagai rujukan keaslian data, sedangkan cloud dipakai untuk simulasi perubahan data (tampering) dan pembuktian deteksi kecurangan.

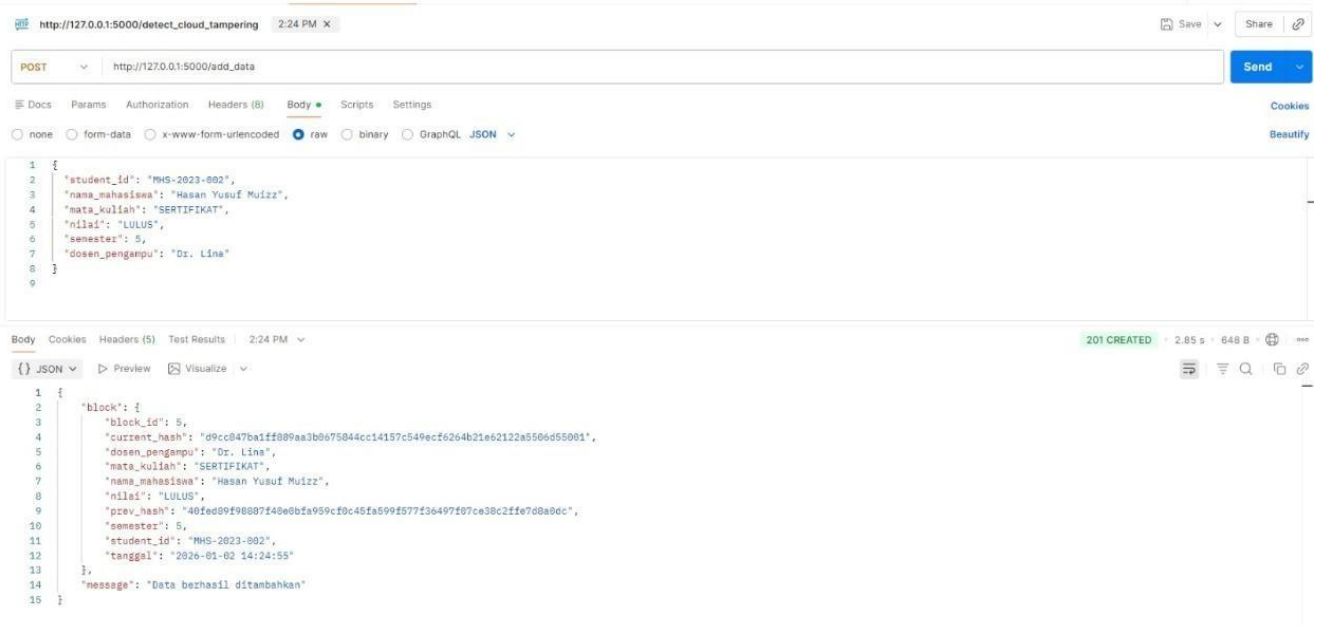
II. ARSITEKTUR UMUM SISTEM

klien mengirim permintaan HTTP ke server Flask API pada `http://127.0.0.1:5000`. Flask memproses request, memvalidasi kelengkapan field, lalu membentuk blok baru pada modul blockchain lokal. Setiap blok memuat data akademik dan link ke blok sebelumnya melalui `prev_hash`, kemudian menghasilkan `current_hash` berbasis SHA-256. Setelah blok terbentuk, sistem juga dirancang untuk mengirimkan data blok tersebut ke Google Sheets melalui endpoint Google Apps Script, sehingga data dapat ditampilkan di cloud dan menjadi objek pengujian manipulasi.

III. TOOLS & ENVIRONMENT

Implementasi menggunakan Python dengan Flask sebagai framework API. Hashing menggunakan SHA-256 melalui pustaka `hashlib`. Pengujian endpoint dilakukan dengan Postman untuk pembuatan data dan permintaan verifikasi. Untuk integrasi cloud, sistem menggunakan Google Apps Script Web App sebagai jembatan HTTP menuju Google Sheets.

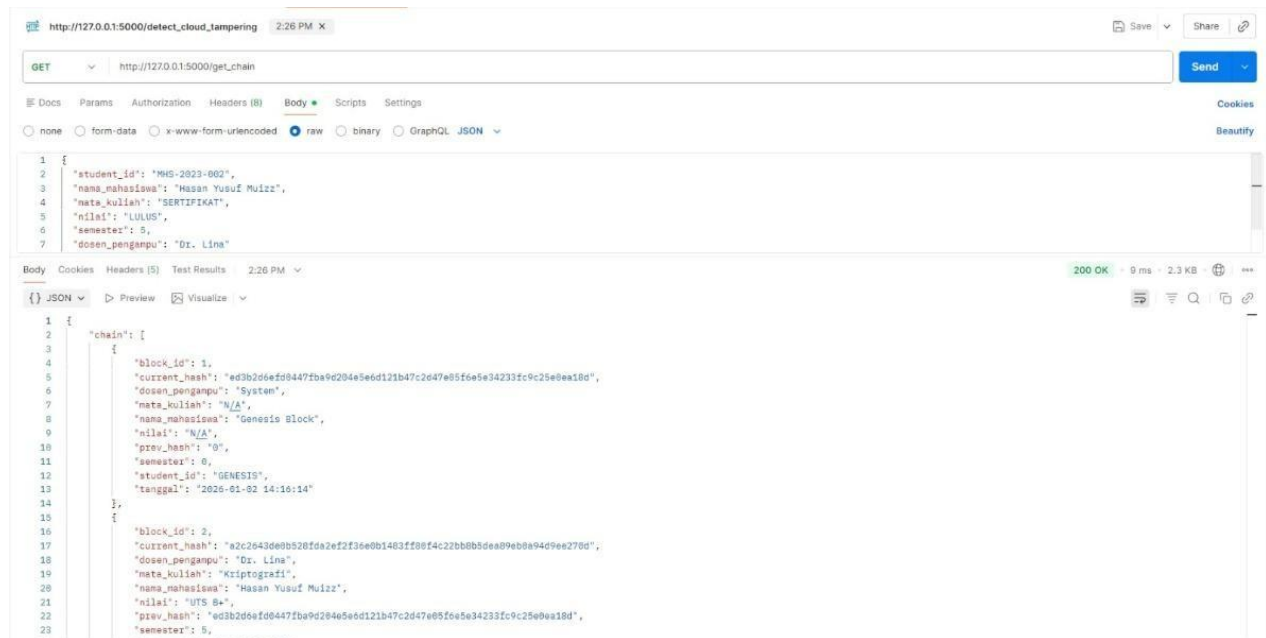
IV. PENGUJIAN DENGAN POSTMAN



Gambar 1. Menginput data lewat POST /add_data

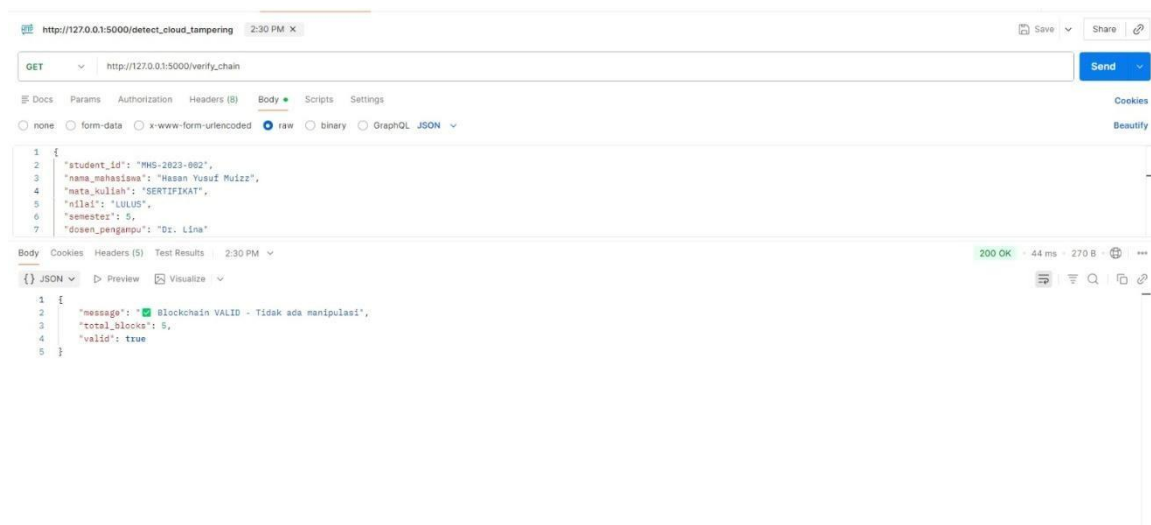
Pengujian diawali dengan eksekusi POST /add_data yang menghasilkan status created. Variabel berisi `student_id: "MHS-2023-002"`, `nama_mahasiswa: "Hasan Yusuf Muizz"`, `mata_kuliah: "SERTIFIKAT"`, `nilai:`

"LULUS", semester: 5, dan dosen_pengampu: "Dr. Lina". Kemudian server merespons dengan mengembalikan objek blok dengan block_id: 5, lengkap dengan current_hash dan prev_hash, serta timestamp blok. Sistem berhasil melakukan pencatatan append-only dan membentuk blok baru yang terhubung ke blok sebelumnya melalui prev_hash.



Gambar 2. Melihat blockchain via GET /get_chain

Setelah penambahan blok, dilakukan GET /get_chain dan server merespons OK dengan daftar rantai blok. Pada respons tersebut terbentuk genesis block sebagai blok awal dengan karakteristik umum: student_id: "GENESIS", nama_mahasiswa: "Genesis Block", mata_kuliah: "N/A", nilai: "N/A", semester: 0, prev_hash: "0", serta timestamp 2026-01-02 14:16:14. Pada blok berikutnya, sistem menampilkan blok akademik untuk MHS2023-002 dengan mata_kuliah: "Kriptografi" dan nilai: "UTS B+", yang membuktikan rantai berisi transaksi akademik bertahap sebelum mencapai status sertifikat.



Gambar 3. Simulasi GET /verify_chain

Tahap verifikasi dilakukan melalui GET /verify_chain dan server merespons OK dengan keluaran valid: true, disertai informasi total_blocks: 5 dan pesan "Blockchain VALID – Tidak ada manipulasi". Hasil ini menyatakan seluruh hubungan prev_hash terhadap current_hash yang merupakan blok sebelumnya memiliki nilai yang konsisten dan hash setiap blok sesuai dengan hasil perhitungan ulang, sehingga blockchain lokal berada pada kondisi benar, tidak ada perubahan pada ledger lokal.

V. SIMULASI KECURANGAN (TAMPERING) DAN ANALISIS SEBELUM - SESUDAH

The image shows a Google Sheet on the left and a REST client on the right. The Google Sheet, titled 'LULUS', contains academic data for a student named Raihan Juniardi. The REST client shows a GET request to 'http://127.0.0.1:5000/detect_cloud_tampering' with a 200 OK response. The response body is a JSON object indicating that the data is safe and matches the local blockchain.

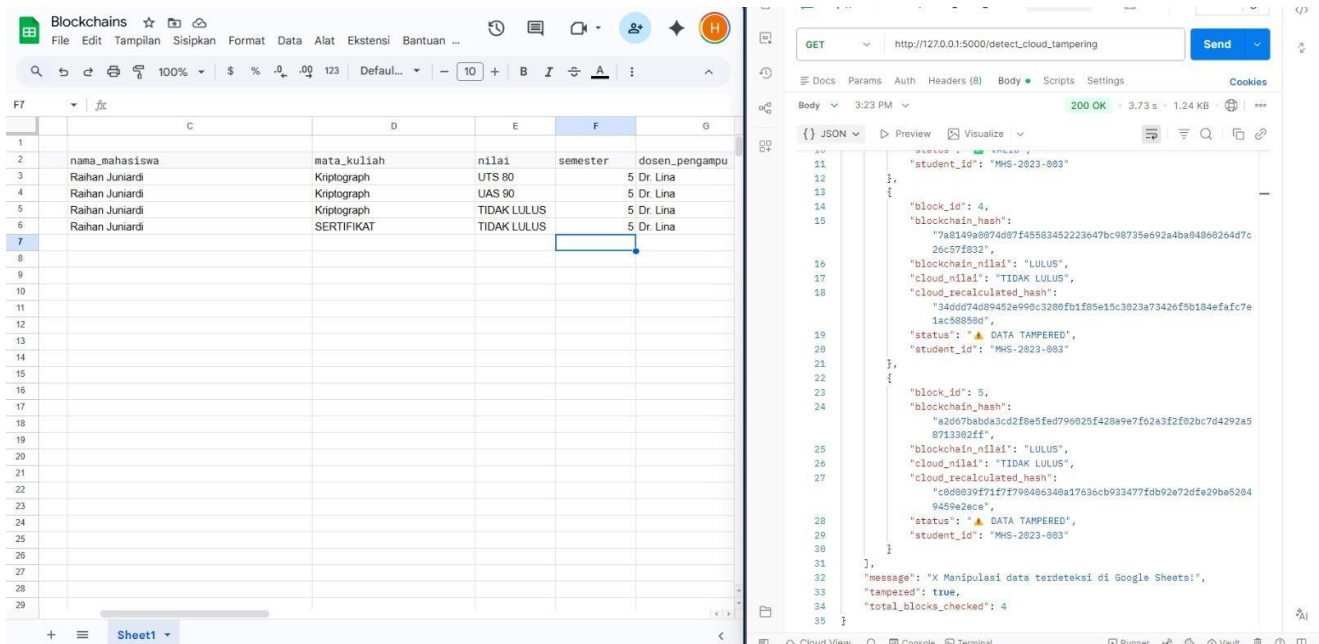
	C	D	E	F	G
1					
2	nama_mahasiswa	mata_kuliah	nilai	semester	dosen_pengampu
3	Raihan Juniardi	UTS 80		5	Dr. Lina
4	Raihan Juniardi	Kriptograph	UAS 90	5	Dr. Lina
5	Raihan Juniardi	Kriptograph	LULUS	5	Dr. Lina
6	Raihan Juniardi	SERTIFIKAT	LULUS	5	Dr. Lina
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					

```

GET http://127.0.0.1:5000/detect_cloud_tampering
200 OK
{
  "student_id": "MHS-2023-003",
  "nama_mahasiswa": "Raihan Juniardi",
  "mata_kuliah": "SERTIFIKAT",
  "nilai": "LULUS",
  "semester": 5,
  "dosen_pengampu": "Dr. Lina"
}
{
  "block_id": 5,
  "status": "VALID",
  "student_id": "MHS-2023-003"
},
{
  "block_id": 4,
  "status": "VALID",
  "student_id": "MHS-2023-003"
},
{
  "block_id": 6,
  "status": "VALID",
  "student_id": "MHS-2023-003"
}
},
{
  "message": "Data cloud aman dan sesuai blockchain",
  "tampered": false,
  "total_blocks_checked": 4
}
  
```

Gambar 4. Sebelum Tampering

Pada kondisi sebelum tampering, data pada Google Sheets masih dengan catatan blockchain lokal. Terlihat dari lembar google sheets yang memuat urutan data akademik atas nama Raihan Juniardi yaitu UTS 80, UAS 90, serta dua entri akhir yang merepresentasikan kelulusan (nilai: LULUS) untuk tahap akhir mata kuliah dan penerbitan sertifikat. Pada saat yang sama, pemanggilan endpoint GET /detect_cloud_tampering menghasilkan respons OK dengan kesimpulan tampered: false dan pesan bahwa data cloud aman serta sesuai blockchain. Sistem juga melaporkan 4 blok diperiksa (total_blocks_checked: 4), dan blok-blok yang relevan dinyatakan VALID, menandakan nilai yang terbaca dari cloud menghasilkan hash yang sama dengan hash acuan yang tersimpan di blockchain lokal.



Gambar 5. Sesudah Tampering

Pada kondisi sesudah tampering, dilakukan manipulasi langsung pada Google Sheets dengan mengubah nilai “LULUS” menjadi “TIDAK LULUS” pada dua entri akhir. Setelah perubahan tersebut, pemanggilan ulang endpoint GET /detect_cloud_tampering tetap merespons OK, namun status berubah menjadi tampered: true dengan pesan “Manipulasi data terdeteksi di Google Sheets”. Pada bagian rincian, sistem menunjukkan adanya ketidaksesuaian pada block_id 4 dan block_id 5 dengan status “DATA TAMPERED”. Bukti ketidaksesuaian dijelaskan melalui perbandingan variabel blockchain_nilai: "LULUS" sedangkan cloud_nilai: "TIDAK LULUS", serta perbedaan antara blockchain_hash dan cloud_recalculated_hash. Perbedaan hash ini merupakan indikator jika isi data di cloud telah berubah dari versi yang sebelumnya dicatat dan dikunci oleh blockchain lokal.

Mekanisme pendeteksian bekerja dengan cara menghitung ulang hash berdasarkan data yang dibaca dari Google Sheets (cloud_recalculated_hash) kemudian membandingkannya dengan hash referensi pada blockchain (blockchain_hash). Karena fungsi hash SHA-256 sangat sensitif terhadap perubahan kecil pada isi data, penggantian satu atribut nilai akan menghasilkan hash yang berbeda. Walaupun blockchain lokal tetap mempertahankan integritas internalnya yaitu append-only dan tidak berubah, sistem tetap mampu mengidentifikasi cloud telah dimanipulasi, sekaligus menunjukkan blok mana yang terdampak beserta bukti nilai sebelum dan sesudah yang tidak sesuai.