

## BAB 3 PELAKSANAAN KERJA MAGANG

### 3.1 Kedudukan dan Organisasi

Pada program kerja magang di Ditjen Dikti terdapat tiga tim yang ditempati oleh mahasiswa magang yaitu tim Metaverse, tim Infrastruktur, dan tim *Blockchain*. Pada pelaksanaan kerja magang ini penulis tergabung dalam tim *Blockchain* dan menempati posisi sebagai *Back-End Developer*. Dalam pelaksanaan program kerja magang, laporan harian mahasiswa magang pada tim *Blockchain* diawasi oleh Bapak Dr.rer.nat. Avinanta Tarigan selaku pembimbing lapangan sekaligus *developer Blockchain*. Kemudian untuk proses pengembangan sistem *Blockchain* diawasi oleh Bapak Andika Demas Riyandi selaku *lead developer* tim *Blockchain* dan juga Ibu Tanti Ruwani, M.Sc. selaku *developer* tim *Blockchain*.

### 3.2 Tugas yang Dilakukan

Selama pelaksanaan kerja magang, tugas dan tanggung jawab yang diberikan kepada mahasiswa magang pada tim *Blockchain* adalah sebagai berikut:

- Memahami alur bisnis dari sistem yang sudah ada sebelumnya.
- Memahami struktur data dari sistem PDDIKTI.
- Mempelajari tentang Web3 dan *Blockchain*.
- Membantu tim *developer* untuk mengembangkan sistem sesuai dengan tugas atau arahan yang diberikan.
- Merancang dan membuat *prototype* situs web *blockchain explorer*.
- Merancang dan membuat *prototype* REST API untuk digunakan pada situs web *blockchain explorer*.

### 3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang berlangsung selama 5 bulan atau sekitar 20 minggu seperti yang diuraikan pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Pada minggu ini penulis mendapatkan informasi-informasi mengenai kegiatan Magang dan Studi Independen Bersertifikat (MSIB) yang diberikan pada acara <i>onboarding</i> nasional.
2	Minggu ini penulis mengikuti kegiatan <i>onboarding</i> internal yang dilaksanakan oleh Dikti dan juga pelatihan oleh Biztech Academy yang dilaksanakan selama 2 hari. Pada pelatihan tersebut penulis belajar mengenai <i>blockchain</i> yang dimulai dari sejarah, teknologi dan arsitektur yang digunakan pada <i>blockchain</i> serta konsensus yang ada pada <i>blockchain</i> . Selain itu penulis juga belajar mengenai <i>metaverse</i> yang dimulai dari konsep <i>metaverse</i> hingga contoh penerapan <i>metaverse</i> pada kasus dunia nyata.
3	Minggu ini penulis belajar tentang dasar-dasar pemrograman pada bahasa pemrograman JavaScript dan Go. Dalam masa pembelajaran, penulis juga mengerjakan latihan-latihan dan tugas yang diberikan oleh <i>trainer</i> penulis. Selain itu penulis juga belajar mengenai API dan membuat REST API sederhana yang menerapkan fungsi CRUD dengan menggunakan bahasa pemrograman Go sekaligus dengan pengujian menggunakan Postman dan juga membuat <i>frontend</i> -nya menggunakan React JS.
4	Minggu ini penulis belajar sekaligus melakukan eksplorasi tentang apa itu <i>blockchain</i> , Web3, dan <i>tech stack</i> yang digunakan. Setiap harinya penulis juga mengikuti <i>daily standup meeting</i> yang membahas tentang <i>progress</i> proyek yang nantinya akan dilakukan serta mempresentasikan tugas yang diberikan kepada mahasiswa magang (tugas <i>flowchart</i> , presentasi proyek mahasiswa). Penulis juga mengikuti rapat di kantor pusat Ditjen Dikti untuk mengetahui dan membahas mengenai proses bisnis, analisis, dan rancangan / keterhubungan dengan sistem yang sudah ada.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
5	<p>Minggu ini penulis berdiskusi bersama tim untuk mencari alur dari sistem PIN yang bisa diterapkan ke dalam sistem <i>blockchain</i>. Setiap harinya penulis juga mengikuti <i>daily meeting</i> yang membahas tentang <i>update progress</i> tim developer dan <i>progress</i> tugas mahasiswa. Penulis juga mengikuti rapat yang membahas tentang sinkronisasi regulasi dengan rancangan <i>smart contract</i> dan rapat yang membahas tentang detail <i>web services</i> yang dibutuhkan untuk PIN dan cara mengaksesnya. Selain itu penulis juga mengerjakan tugas yang diberikan mengenai pembuatan aplikasi Web3 dan <i>smart contract</i> serta mengikuti <i>workshop</i> tentang <i>blockchain</i>.</p>
6	<p>Minggu ini penulis mempelajari lebih dalam tentang DeFi Kingdoms seperti membaca dokumentasinya dan mencobanya secara langsung. Lalu penulis juga belajar tentang The Graph dan mempresentasikannya pada <i>daily meeting</i>. Kemudian penulis juga membuat <i>sequence diagram</i> dari sistem PIN dan juga aplikasi <i>blockchain</i> yang akan dibangun serta membuat UI <i>prototype</i> dari aplikasi <i>blockchain</i> tersebut menggunakan Figma. Terakhir, penulis juga berdiskusi bersama dengan tim magang untuk mempelajari dan membahas tentang MakerDAO.</p>
7	<p>Selain mengikuti <i>daily meeting</i>, minggu ini penulis mempelajari cara membuat NFT Marketplace dan juga ikut membantu membuat dua <i>flow smart contract</i> dari NFT Marketplace bersama dengan tim mahasiswa. Penulis juga membuat <i>mockup</i> dan <i>prototype</i> dari proses ketika mahasiswa ingin melakukan klaim NFT. Kemudian penulis juga ikut membuat alur UI dari <i>mockup</i> tersebut. Selain itu penulis bersama dengan tim mahasiswa magang juga berdiskusi mengenai <i>flow</i> dari proses yang nantinya akan diterapkan ke aplikasi dan membuat <i>sequence diagram</i>-nya.</p>
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
8	Selain mengikuti <i>daily meeting</i> , minggu ini penulis membuat dokumen dokumentasi yang berisi penjelasan dari setiap <i>sequence diagram</i> yang telah dibuat. Kemudian penulis juga membuat <i>blockchain explorer</i> dengan menggunakan Next.js untuk <i>frontend</i> dan Express.js untuk <i>backend</i> (API). Untuk mengambil data dari <i>blockchain</i> , digunakan <i>library</i> web3.js dengan <i>node provider</i> Infura dan menggunakan <i>testnet</i> Avalanche sebelum akhirnya berhasil menggunakan RPC Dikti.
9	Minggu ini penulis menambahkan beberapa fitur atau <i>endpoint</i> baru pada API yang sudah dibuat sebelumnya dengan menggunakan Express.js. Kemudian penulis juga mencoba melakukan instalasi Ceramic mulai dari Ceramic CLI hingga Ceramic HTTP Client serta mencoba <i>repository</i> GitHub tentang sistem autentikasi dengan menggunakan Ceramic dan Self.ID. Lalu penulis juga mengikuti rapat harian yang membahas tentang <i>update progress</i> tim developer & mahasiswa serta rapat dengan tim infrastruktur untuk membahas IPFS.
10	Minggu ini penulis mengikuti rapat di kantor Ditjen Diktiristek untuk berdiskusi tentang layanan pendidikan tinggi dalam rangka pengembangan proyek yang sedang dilakukan. Kemudian penulis juga mengikuti dua rapat bersama dengan tim divisi lain untuk membahas tentang <i>progress</i> dan kesiapan dari masing-masing divisi. Lalu penulis juga membahas bersama dengan tim mahasiswa magang terkait <i>Decentralized Identifiers</i> , membuat diagram-nya, dan eksplorasi <i>wallet</i> MetaMask. Selain itu, setiap harinya penulis juga mengikuti <i>daily meeting</i> yang membahas tentang <i>update progress</i> dari tim developer.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
11	<p>Minggu ini penulis mengikuti rapat koordinasi antara beberapa tim divisi lain untuk membahas tentang <i>progress</i> dan kesiapan dari masing-masing tim. Kemudian penulis juga mengikuti rapat singkat pada pagi hari yang membahas tentang apa yang dikerjakan oleh mahasiswa pada hari sebelumnya dan hal-hal yang akan dilakukan mahasiswa pada hari tersebut. Selain itu penulis juga menghadiri acara presentasi finalis lomba Metaversitas yang dilaksanakan di Yogyakarta serta membuat notula dari acara tersebut. Pada hari berikutnya penulis juga mengikuti acara seminar Merdeka Belajar di Metaversitas yang terdiri dari tiga tema seminar serta membuat notulanya. Pada acara tersebut juga terdapat acara pameran dari proyek Metaversitas yang dikerjakan serta pengumuman lomba Metaversitas. Kemudian penulis juga membuat laporan bulanan yang berisi tugas-tugas yang telah dilakukan pada bulan ini.</p>
12	<p>Minggu ini, selain mengikuti rapat harian, penulis juga mengerjakan beberapa hal yang meliputi pembuatan dokumentasi mengenai DID dan portal Metaversitas, membuat <i>flowchart</i> dan <i>sequence diagram</i> dari proses pembuatan DID dan NFT, serta <i>mockup &amp; prototype</i> dari alur <i>login</i> dan pembuatan NFT pada portal Metaversitas.</p>
13	<p>Minggu ini penulis melakukan revisi pada <i>mockup</i> portal Metaversitas yang sudah dibuat sebelumnya pada Figma dan juga membuat alur <i>prototype</i>-nya. Selain itu penulis juga ikut membantu untuk membuat algoritma dari <i>smart contract</i>. Kemudian penulis juga mengikuti rapat harian yang membahas mulai dari tugas yang akan dilakukan oleh mahasiswa dan juga <i>update progress</i> dari tim developer dan mahasiswa.</p>
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
14	Selain mengikuti <i>daily standup meeting</i> , penulis mengikuti rapat koordinasi antara tim <i>blockchain</i> dan <i>metaverse</i> . Selain itu penulis juga mengikuti rapat yang membahas tentang tugas yang akan dikerjakan oleh mahasiswa magang. Kemudian penulis juga mengerjakan beberapa hal seperti membuat dokumentasi yang berupa UML mengenai protokol dan juga struktur data API, revisi <i>smart contract</i> , dan mencoba untuk mengoptimalkan <i>blockchain explorer</i> .
15	Selain mengikuti rapat yang dilakukan setiap harinya, minggu ini penulis bersama dengan tim magang berdiskusi untuk membahas konten apa yang diperlukan pada halaman web. Kemudian penulis juga membuat <i>mockup</i> dan <i>prototype landing page</i> serta memasukkan konten yang telah didiskusikan sebelumnya pada <i>landing page</i> . Selain itu penulis juga membuat <i>list</i> berupa tabel yang berisi data-data yang dibutuhkan yang nantinya akan dijadikan basis data.
16	Selain mengikuti rapat harian, minggu ini penulis mengerjakan beberapa hal yaitu revisi <i>sequence diagram</i> protokol aplikasi, <i>refactoring</i> kode pada <i>frontend blockchain explorer</i> serta menambahkan fitur <i>pagination</i> , fitur <i>query</i> kolom spesifik pada API <i>blockchain explorer</i> , <i>deployment</i> perubahan pada <i>frontend</i> dan juga <i>backend</i> ke server serta melakukan perubahan bahasa pada <i>blockchain explorer</i> menjadi bahasa Indonesia.
17	Minggu ini penulis kembali mempelajari tentang teknologi-teknologi baru yang penulis temukan selama proyek ini berlangsung. Selain itu penulis juga mempelajari kembali tentang proyek yang dikerjakan di program magang ini.
18	Minggu ini penulis mempelajari kembali tentang teknologi dan teori yang berkaitan dengan <i>blockchain</i> . Selain itu penulis juga kembali merangkum tugas dan proyek yang telah dikerjakan selama magang sebagai bahan persiapan penulis untuk mengerjakan laporan magang nantinya.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
19	Minggu ini penulis mengerjakan laporan akhir magang yang diperlukan untuk program kampus merdeka.
20	Minggu ini penulis mengerjakan laporan akhir magang yang diperlukan untuk kampus.

### 3.3.1 Alat yang digunakan

Dalam pengembangan situs web *blockchain explorer*, diperlukan beberapa perangkat lunak pendukung yang terdiri dari:

- Visual Studio Code v1.74.3
- XAMPP v8.0.12-0
- Postman v10.5.6
- Google Chrome v109.0.5414.75
- DBeaver v22.2.5.202211202223
- FileZilla v3.62.2

### 3.3.2 Struktur Basis Data

Basis data yang digunakan pada situs web *blockchain explorer* ini adalah MySQL yang terdiri dari tiga tabel yaitu tabel *blocks*, tabel *transactions*, dan tabel *addresses*. Berikut merupakan struktur basis data yang digunakan pada situs web *blockchain explorer* ini.

## A. Tabel *Blocks*

blocks	
PK	<u>id INT AUTO_INCREMENT</u>
	number INT UNIQUE
	hash VARCHAR UNIQUE
	parentHash VARCHAR NOT NULL
	baseFeePerGas INT NOT NUL
	nonce VARCHAR
	sha3Uncles VARCHAR NOT NULL
	logsBloom LONGTEXT
	transactionsRoot VARCHAR NOT NULL
	stateRoot VARCHAR NOT NULL
	miner VARCHAR NOT NULL
	difficulty VARCHAR NOT NULL
	totalDifficulty VARCHAR NOT NULL
	extraData VARCHAR NOT NULL
	size INT NOT NULL
	gasLimit INT NOT NULL
	gasUsed INT NOT NULL
	timestamp INT NOT NULL

Gambar 3.1. Struktur basis data tabel *blocks*

Pada Gambar 3.1 merupakan struktur basis data dari tabel bernama *blocks* yang berfungsi untuk menyimpan semua data yang terkait dengan informasi suatu *blocks* yang ada di dalam jaringan *blockchain*.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## B. Tabel *Transactions*

transactions	
PK	<u>id INT AUTO INCREMENT</u>
	hash VARCHAR NOT NULL UNIQUE
	nonce INT NOT NULL
	blockHash VARCHAR
	blockNumber INT
	transactionIndex INT
	from VARCHAR NOT NULL
	to VARCHAR
	contractAddress VARCHAR
	value VARCHAR NOT NULL
	gasPrice VARCHAR NOT NULL
	gas INT NOT NULL
	input LONGTEXT NOT NULL

Gambar 3.2. Struktur basis data tabel *transactions*

Pada Gambar 3.2 merupakan struktur basis data dari tabel bernama *transactions* yang berfungsi untuk menyimpan semua informasi yang berkaitan dengan suatu transaksi yang terdapat di dalam blok pada jaringan *blockchain*.

## C. Tabel *Addresses*

addresses	
PK	<u>id INT AUTO INCREMENT</u>
	address VARCHAR NOT NULL UNIQUE
	balance VARCHAR NOT NULL DEFAULT '0'
	isContract BOOLEAN NOT NULL DEFAULT FALSE

Gambar 3.3. Struktur basis data tabel *addresses*

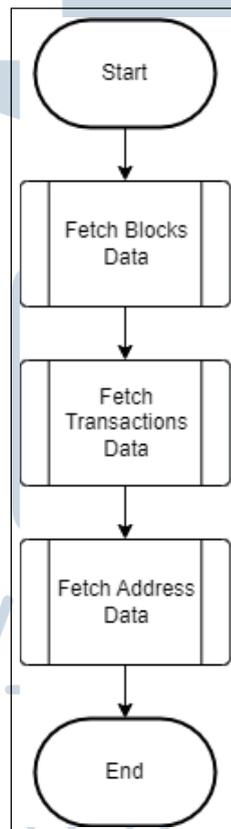
Pada Gambar 3.3 merupakan struktur basis data dari tabel bernama *addresses* yang berfungsi untuk menyimpan semua data yang terkait dengan informasi suatu *address* atau alamat yang ada di dalam jaringan *blockchain*.

### 3.3.3 Flowchart

Pada pengembangan situs web *blockchain explorer* terdapat beberapa hal yang penulis kerjakan, mulai dari pembuatan API, pembuatan tampilan situs web *blockchain explorer*, hingga integrasi API yang sudah dibuat ke situs web *blockchain explorer*. Dalam pembuatan API, *framework* yang digunakan adalah Express.js versi 4.18.1 serta *library* web3.js versi 1.8.0 yang berfungsi untuk melakukan interaksi dengan *node blockchain*. Lalu untuk pembuatan *frontend* situs web *blockchain explorer*, *framework* yang digunakan adalah Next.js versi 12.3.1.

Berikut merupakan *flowchart* terkait *blockchain explorer* yang terdiri dari empat halaman dan satu fitur pencarian yang terdapat pada situs web, serta fitur *scheduler*.

#### A. Fitur Scheduler

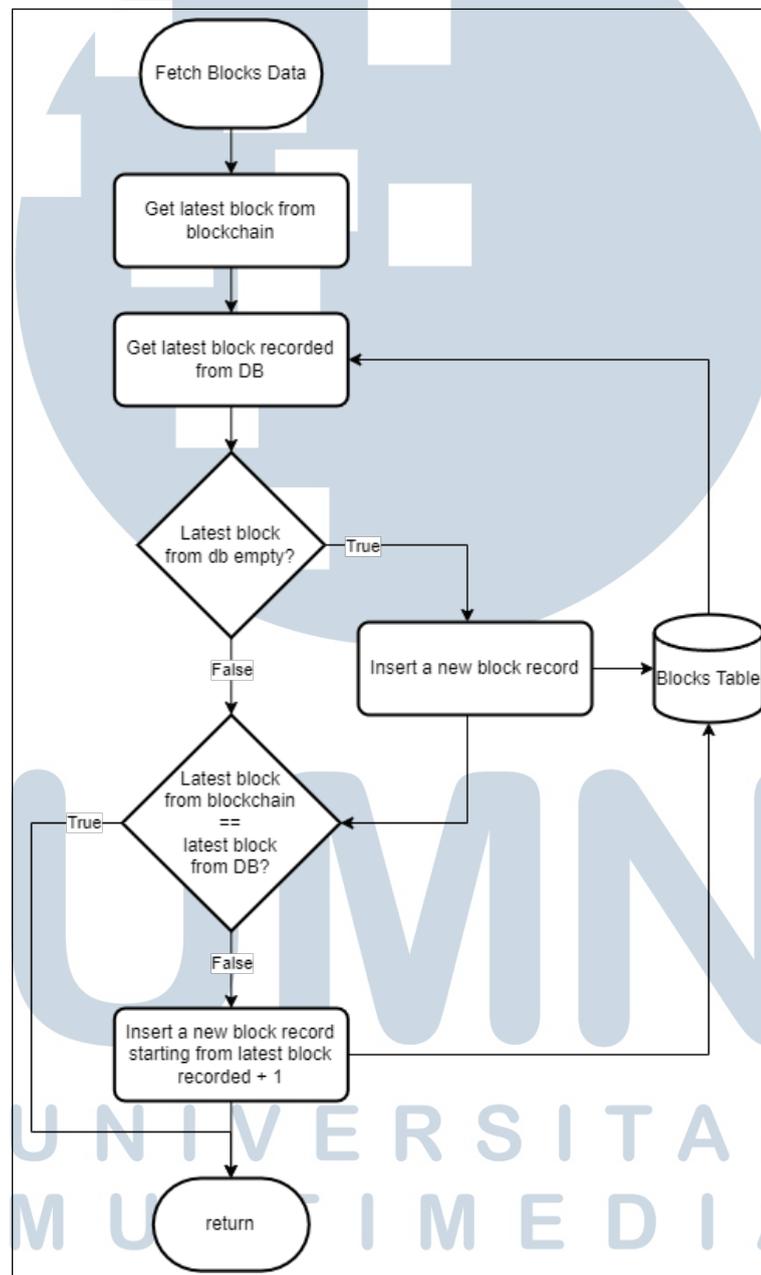


Gambar 3.4. Diagram fitur *scheduler*

Pada Gambar 3.4 merupakan alur dari fitur *scheduler* yang berfungsi untuk mengambil data-data dari blockchain berdasarkan rentang waktu tertentu. Berikut

ini adalah detail alur dari setiap proses yang terdapat di dalamnya.

### A.1 *Fetch Blocks Data*



Gambar 3.5. Diagram proses pengambilan data blok pada fitur *scheduler*

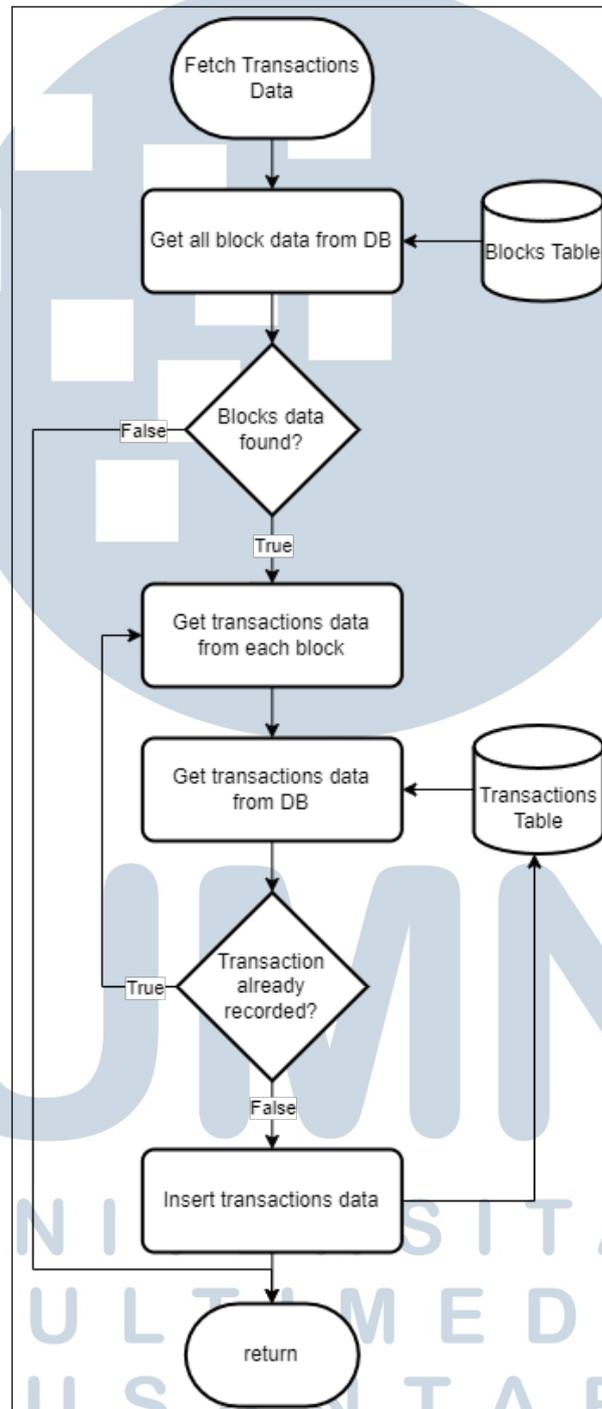
Pada Gambar 3.5 merupakan alur dari proses pengambilan data blok pada *blockchain* yang dimulai dari mengambil data blok terakhir pada *blockchain* dengan menggunakan *library* *web3.js*. Selain itu, program juga akan mengambil data blok

terakhir yang sudah tercatat sebelumnya pada *database*. Jika data blok terakhir pada *database* tidak ditemukan alias kosong, maka program akan memasukkan data blok baru yang didapatkan dengan menggunakan *library* web3.js ke *database*.

Setelah itu, program akan membandingkan apakah data blok terakhir pada *blockchain* sama dengan data blok terakhir yang sudah tercatat pada *database*. Jika data tersebut sama, maka berarti data blok yang tercatat pada *database* sudah merupakan data terbaru yang terdapat pada *blockchain*. Namun jika tidak, maka program akan memasukkan data blok baru pada *database* yang dimulai dari data blok terakhir yang sudah tercatat pada *database* sampai dengan data blok terakhir yang ada pada *blockchain*.



## A.2 Fetch Transactions Data



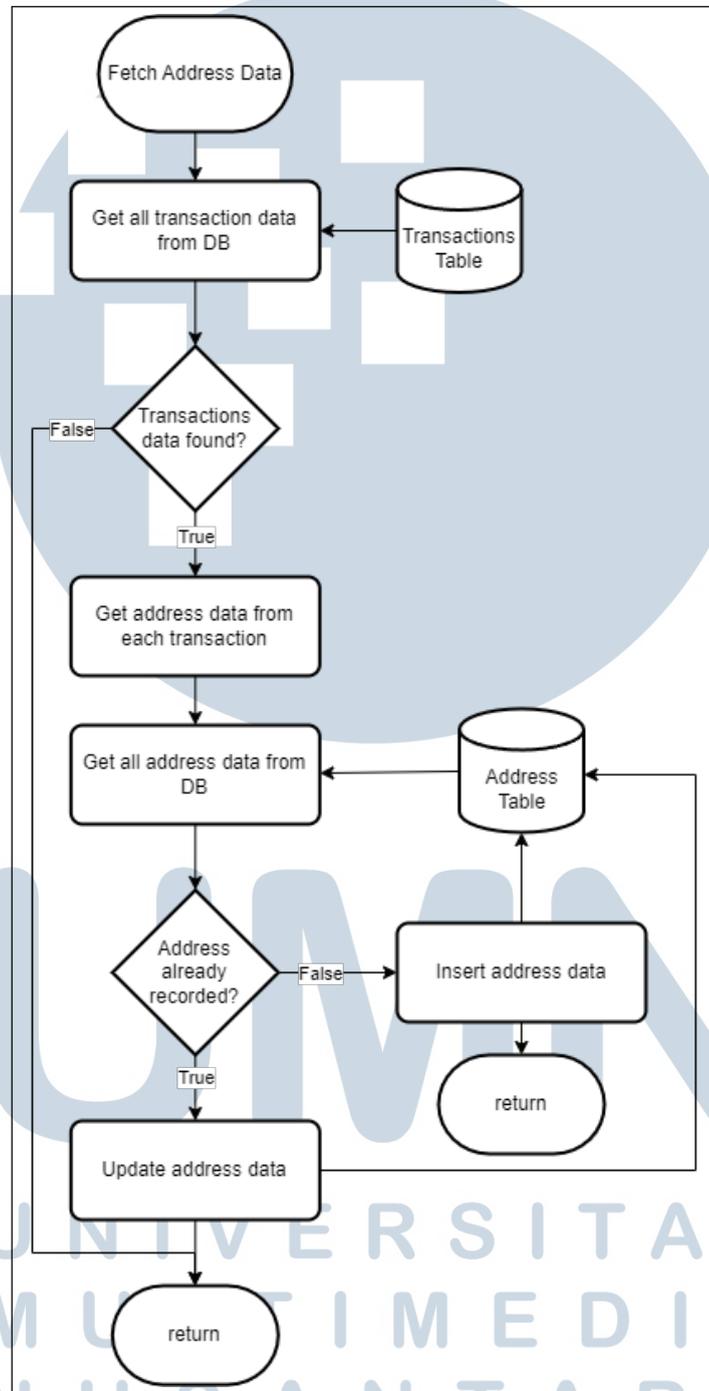
Gambar 3.6. Diagram proses pengambilan data transaksi pada fitur *scheduler*

Pada Gambar 3.6 merupakan alur dari proses pengambilan data transaksi pada *blockchain* yang dimulai dari mengambil semua data blok pada *database*. Jika

data blok ditemukan, maka program akan mengambil data transaksi pada setiap blok dengan menggunakan *library* web3.js dan juga mengambil data transaksi yang sudah tercatat sebelumnya pada *database*. Setelah itu, program akan memeriksa apakah data transaksi tersebut sudah tercatat sebelumnya pada *database*. Jika sudah tercatat, maka program akan lanjut pada proses mengambil data transaksi di blok berikutnya. Namun jika belum tercatat, maka program akan memasukkan data transaksi tersebut ke dalam *database*.



### A.3 Fetch Address Data

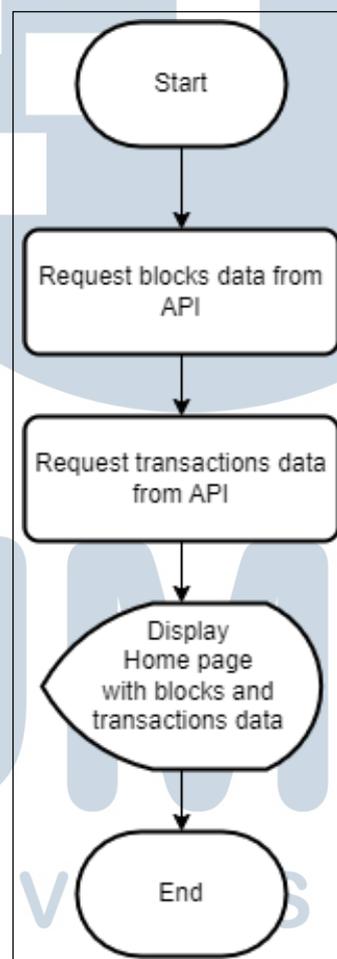


Gambar 3.7. Diagram proses pengambilan data alamat pada fitur *scheduler*

Pada Gambar 3.7 merupakan alur dari proses pengambilan data alamat pada *blockchain* yang dimulai dari mengambil semua data transaksi pada *database*. Jika data transaksi ditemukan, maka program akan mengambil data alamat pada

setiap transaksi dengan menggunakan *library* web3.js dan juga mengambil data alamat yang sudah tercatat sebelumnya pada *database*. Setelah itu, program akan memeriksa apakah data alamat tersebut sudah tercatat sebelumnya pada *database*. Jika belum tercatat, maka program akan memasukkan data alamat tersebut ke dalam *database*. Namun jika sudah tercatat, maka program akan melakukan *update* pada data alamat tersebut dan menyimpannya kembali ke dalam *database*.

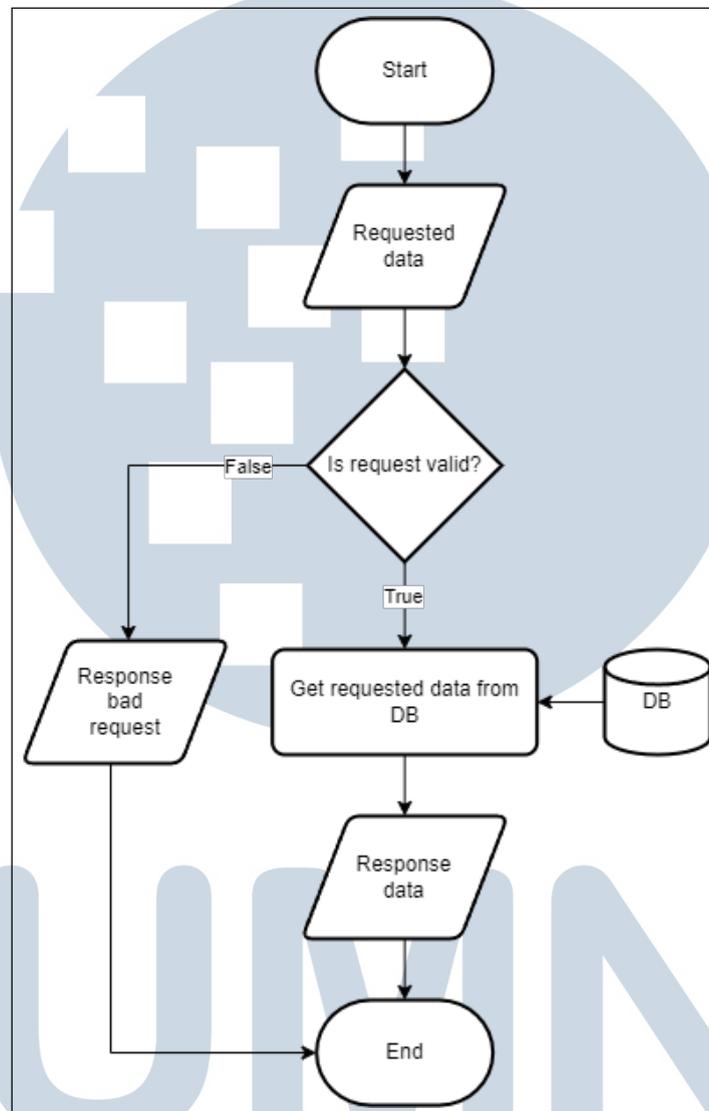
## B. Halaman Utama



Gambar 3.8. Diagram halaman utama pada situs web *blockchain explorer*

Pada Gambar 3.8 merupakan alur dari halaman utama di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint blocks* dan *endpoint transactions* dengan beberapa *query* parameter yaitu *amount* dan *field*. *Query amount* berfungsi untuk menentukan berapa jumlah data yang akan diambil dan

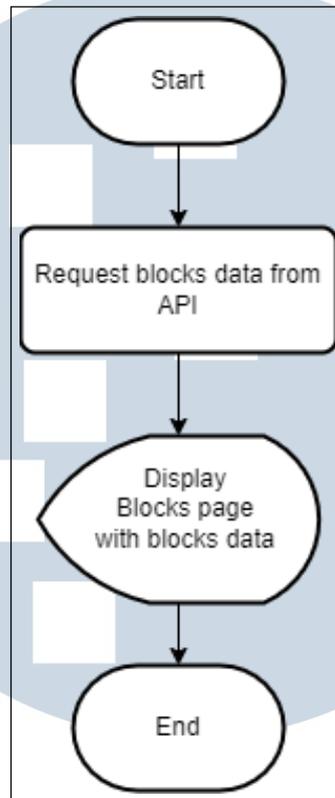
*query fields* berfungsi untuk menentukan data pada kolom apa saja yang akan diambil.



Gambar 3.9. Diagram proses validasi *request* pada API *blockchain explorer*

Pada Gambar 3.9 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

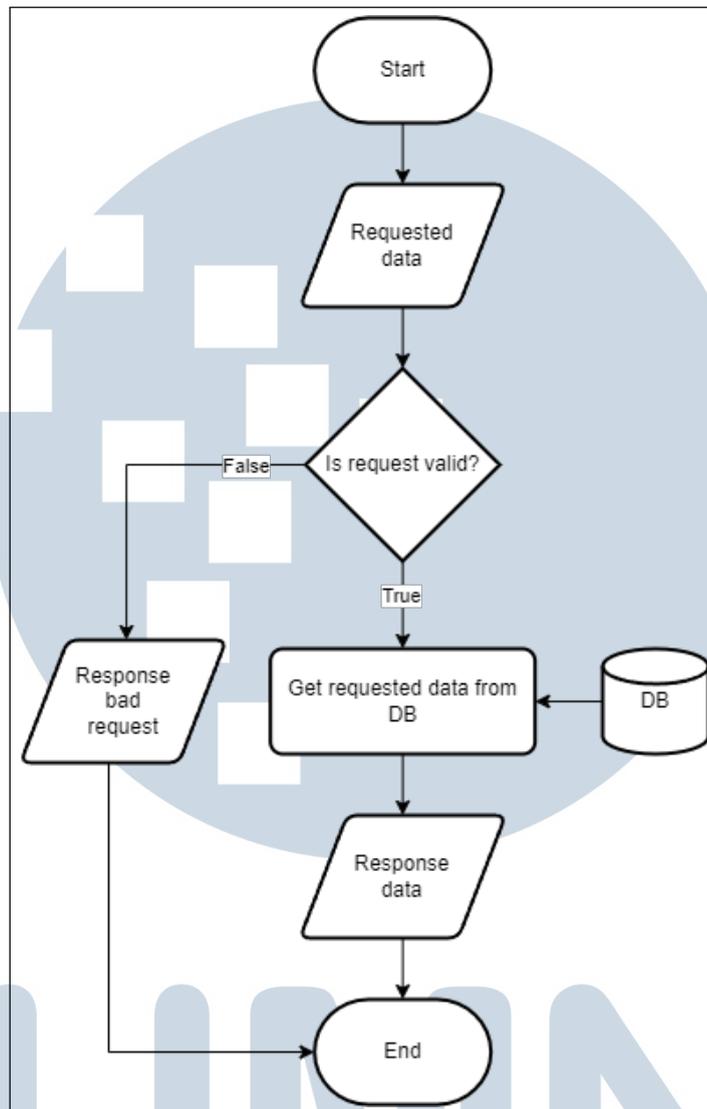
### C. Halaman Daftar Blok



Gambar 3.10. Diagram halaman daftar blok pada situs web *blockchain explorer*

Pada Gambar 3.10 merupakan alur dari halaman daftar blok di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint blocks* dengan *query* parameter berupa *fields* yang berfungsi untuk menentukan data pada kolom apa saja yang akan diambil.

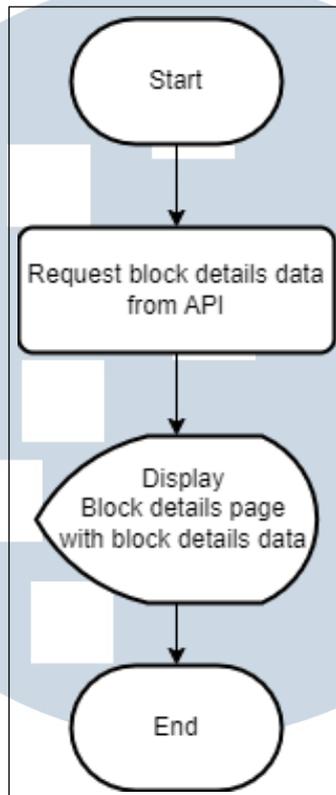
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.11. Diagram proses validasi *request* pada API *blockchain explorer*

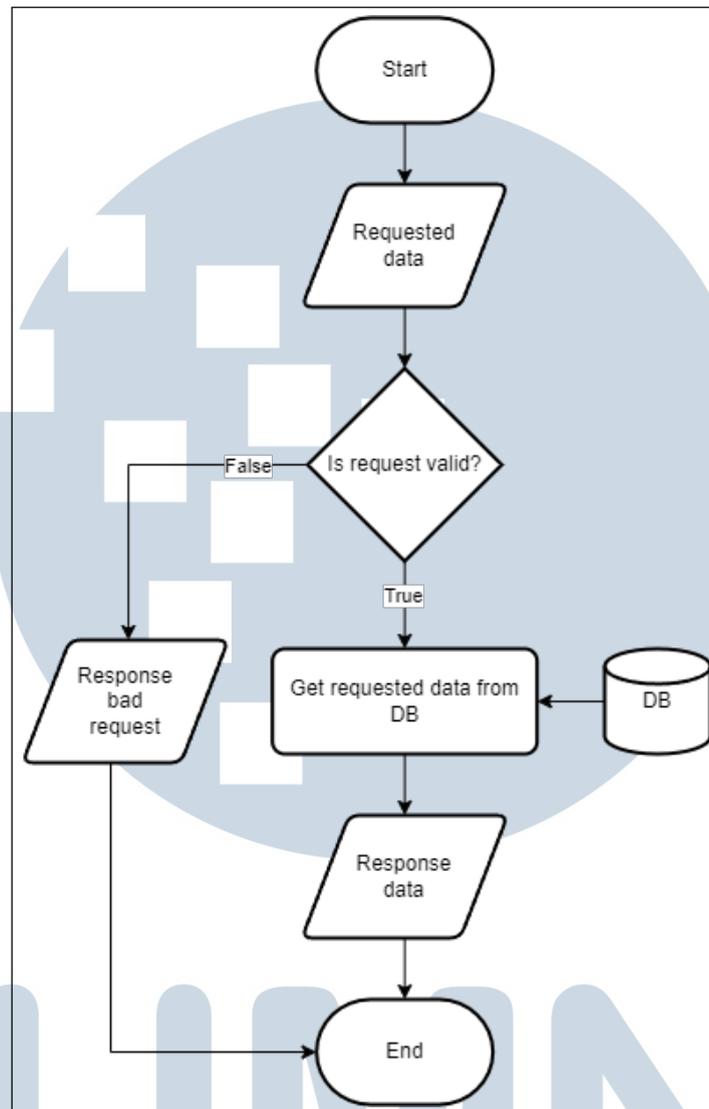
Pada Gambar 3.11 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

#### D. Halaman Detail Blok



Gambar 3.12. Diagram halaman detail blok pada situs web *blockchain explorer*

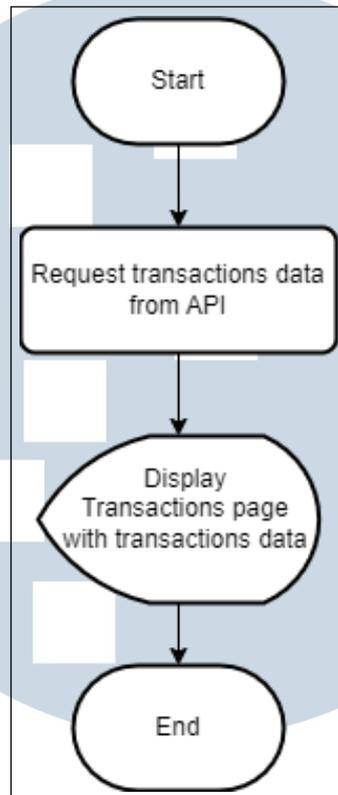
Pada Gambar 3.12 merupakan alur dari halaman detail blok di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint blocks* dengan tambahan parameter berupa nomor atau hash dari suatu blok. Parameter tersebut digunakan untuk menentukan blok mana yang akan diambil detailnya. Selain itu, situs web *blockchain explorer* juga akan melakukan *request* HTTP dengan metode GET pada *endpoint* dan parameter yang sama pada *request* sebelumnya, namun dengan tambahan parameter *transactions* yang berfungsi untuk mengambil daftar transaksi yang terdapat pada nomor atau hash blok tersebut.



Gambar 3.13. Diagram proses validasi *request* pada API *blockchain explorer*

Pada Gambar 3.13 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

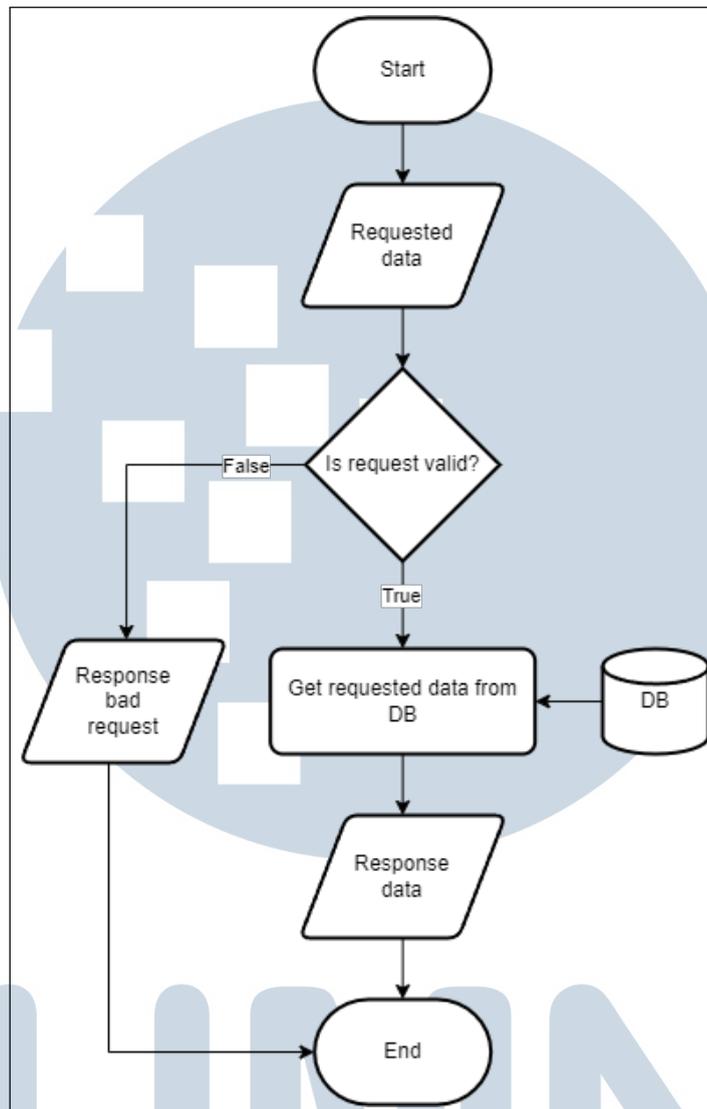
## E. Halaman Daftar Transaksi



Gambar 3.14. Diagram halaman daftar transaksi pada situs web *blockchain explorer*

Pada Gambar 3.14 merupakan alur dari halaman daftar transaksi di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint transactions* dengan *query* parameter berupa *fields* yang berfungsi untuk menentukan data pada kolom apa saja yang akan diambil.

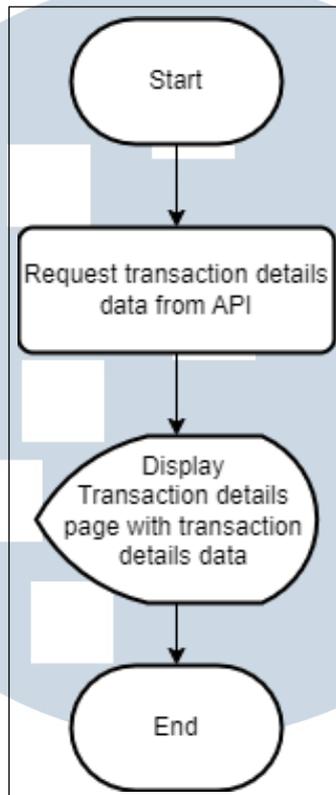
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.15. Diagram proses validasi *request* pada API *blockchain explorer*

Pada Gambar 3.15 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

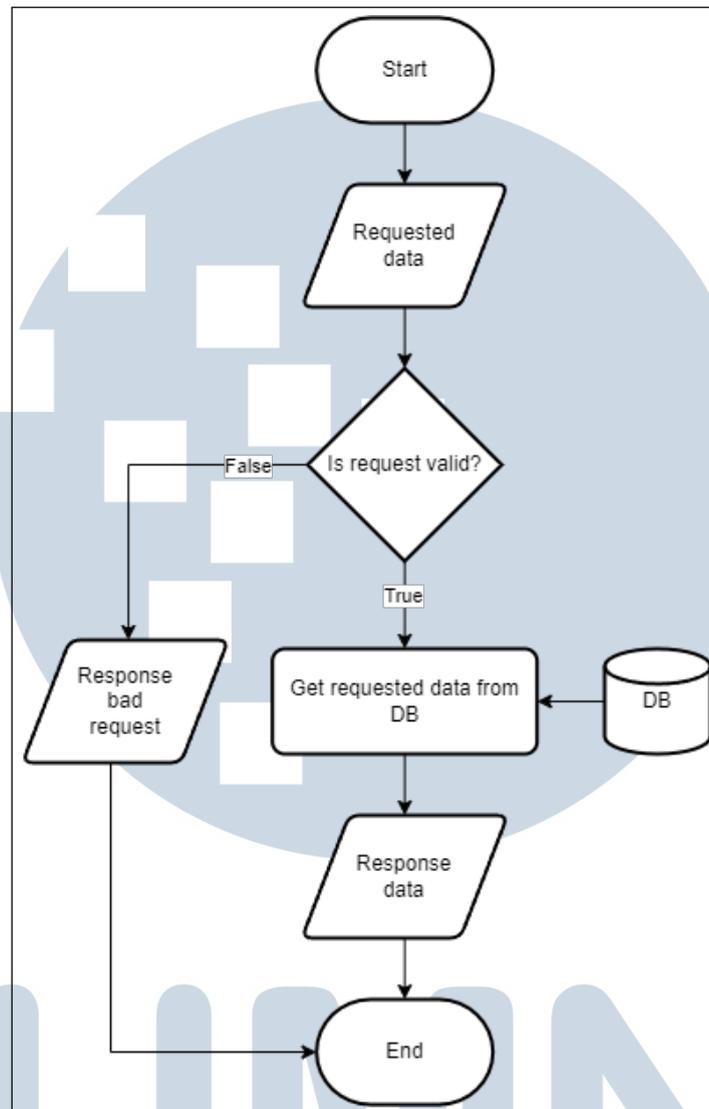
## E. Halaman Detail Transaksi



Gambar 3.16. Diagram halaman detail transaksi pada situs web *blockchain explorer*

Pada Gambar 3.16 merupakan alur dari halaman detail transaksi di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint transactions* dengan tambahan parameter berupa hash dari suatu transaksi. Parameter tersebut digunakan untuk menentukan transaksi mana yang akan diambil detailnya.

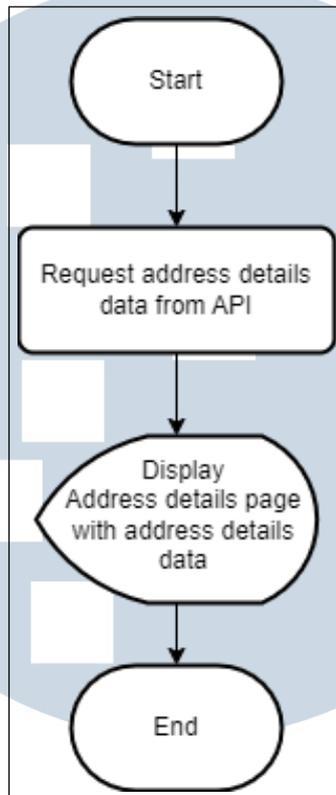
U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.17. Diagram proses validasi *request* pada API *blockchain explorer*

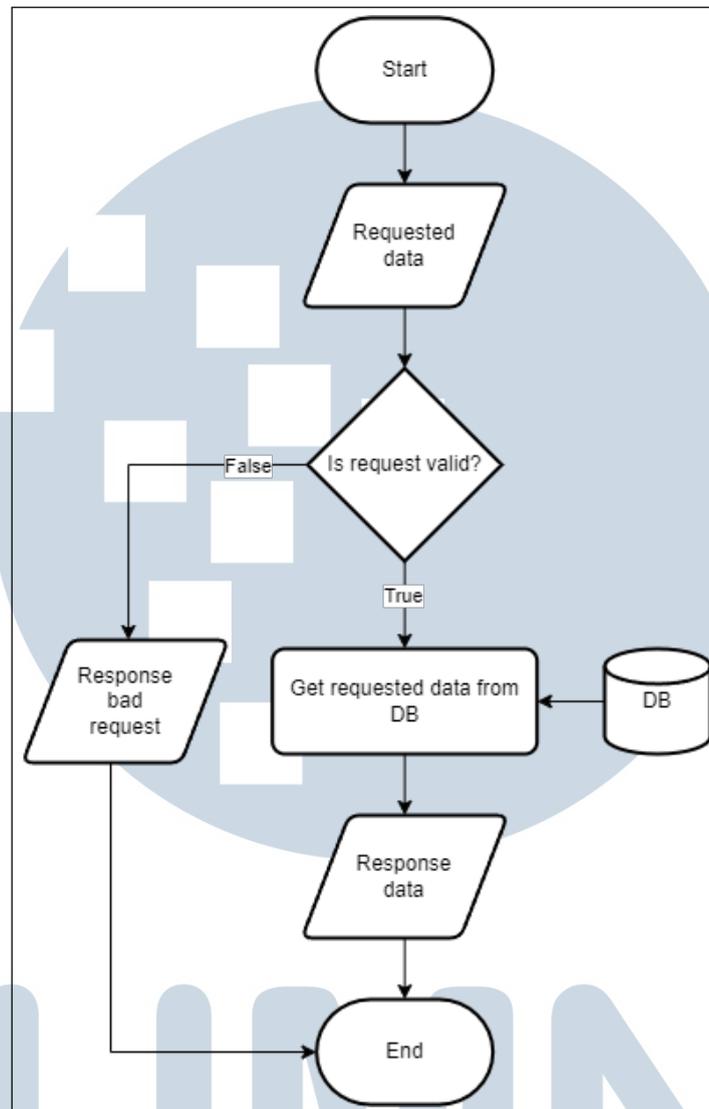
Pada Gambar 3.17 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

## G. Halaman Detail Alamat



Gambar 3.18. Diagram halaman detail alamat pada situs web *blockchain explorer*

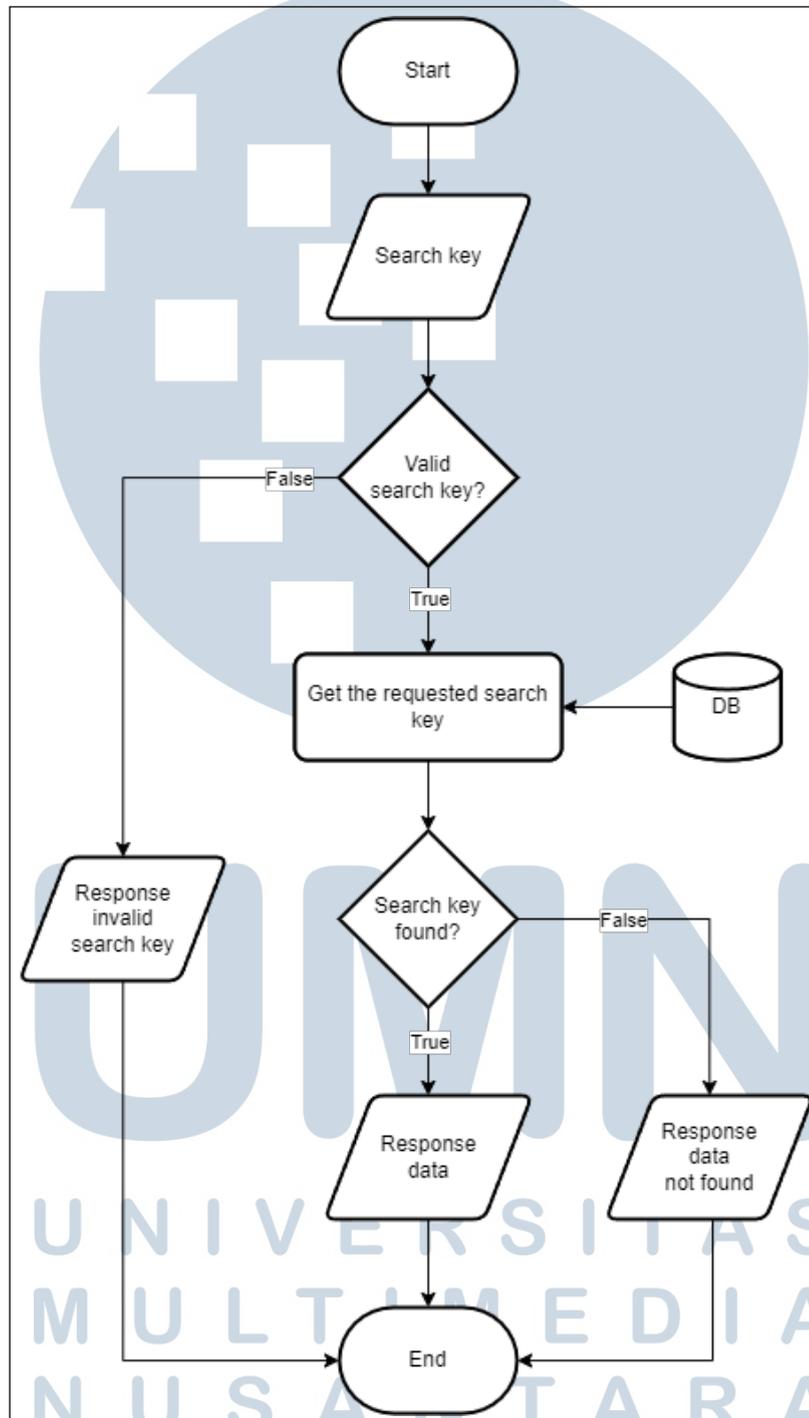
Pada Gambar 3.18 merupakan alur dari halaman detail *address* di mana ketika pengguna mengakses halaman tersebut, situs web *blockchain explorer* akan melakukan *request* HTTP dengan metode GET pada *endpoint addresses* dengan tambahan parameter berupa *address* atau alamat yang diinginkan. Parameter tersebut digunakan untuk menentukan *address* atau alamat mana yang akan diambil detailnya. Selain itu, situs web *blockchain explorer* juga akan melakukan *request* HTTP dengan metode GET pada *endpoint* dan parameter yang sama pada *request* sebelumnya, namun dengan tambahan parameter *transactions* yang berfungsi untuk mengambil daftar transaksi yang terdapat *address* atau alamat tersebut.



Gambar 3.19. Diagram proses validasi *request* pada API *blockchain explorer*

Pada Gambar 3.19 merupakan alur dari proses validasi *request* pada API. Setiap *request* akan melewati proses validasi input pada API untuk menentukan apakah *request* yang dikirimkan valid atau tidak. Jika *request* valid, maka API akan memproses *request* tersebut dan mengembalikan *response* berupa data yang diinginkan dari *database*. Namun jika *request* tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error*.

## H. Fitur Pencarian



Gambar 3.20. Diagram fitur pencarian pada situs web *blockchain explorer*

Pada Gambar 3.20 merupakan alur dari fitur pencarian yang terdapat pada API dengan *endpoint search* di mana pada *endpoint* tersebut terdapat *query*

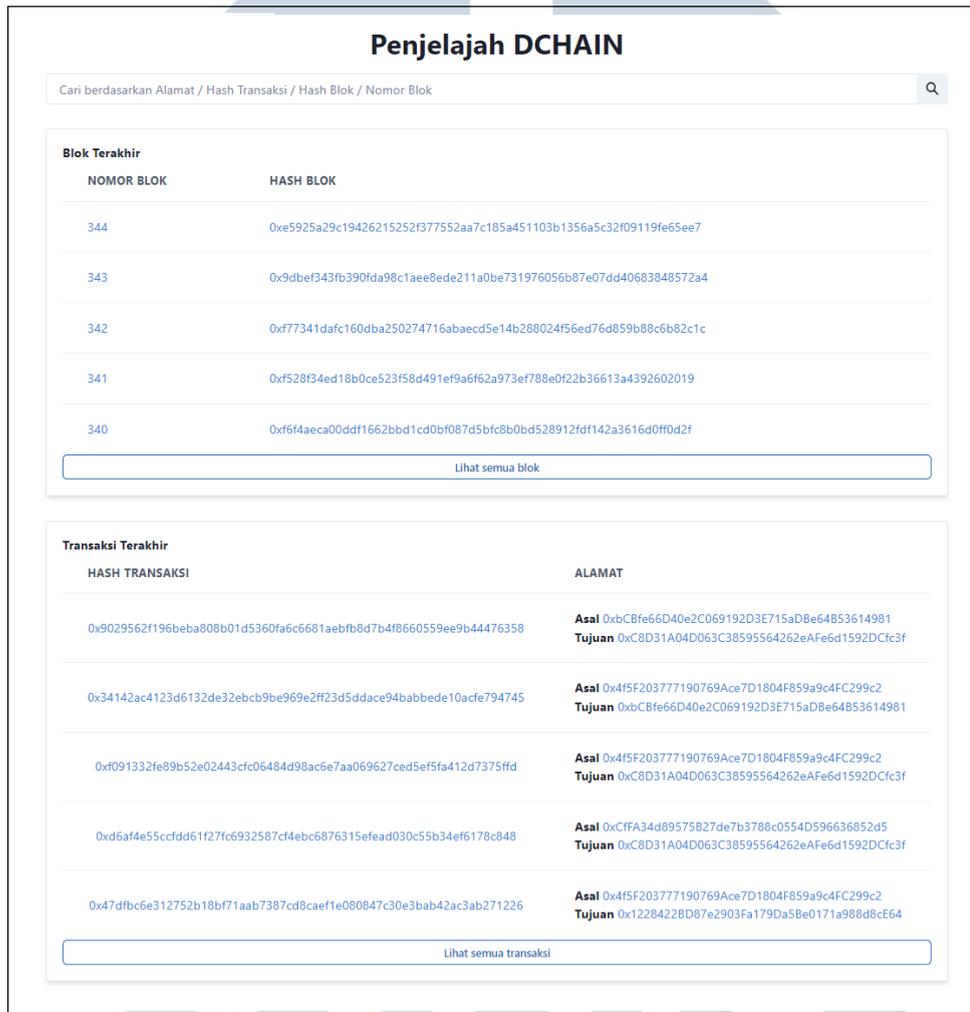
parameter berupa *key* yang berfungsi untuk menentukan data yang akan dicari. Ketika *endpoint* pada API tersebut menerima sebuah input berupa *key* atau data yang akan dicari, API akan melakukan validasi pada input tersebut terlebih dahulu untuk menentukan apakah input yang diterima valid atau tidak. Jika input tidak valid, maka API akan mengembalikan *response* yang berupa pesan *error* bahwa input yang diterima tidak valid.

Ketika input yang diterima valid, maka API akan melakukan proses berikutnya yaitu mencari input *key* tersebut pada *database*. Jika input *key* yang diberikan dapat ditemukan pada *database*, maka API akan mengembalikan *response* yang berupa data yang dicari. Namun jika tidak ditemukan, maka API akan mengembalikan *response* yang berupa pesan yang menyatakan bahwa data yang dicari tidak ditemukan.



### 3.3.4 Hasil Implementasi

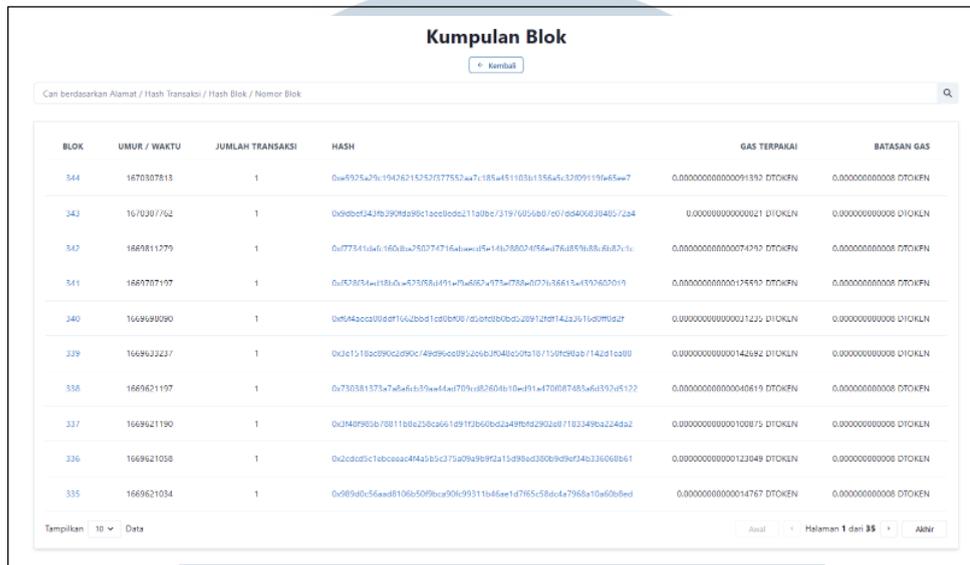
#### A. Halaman Utama



Gambar 3.21. Halaman utama pada situs web *blockchain explorer*

Pada Gambar 3.21 merupakan tampilan halaman utama dari situs web *blockchain explorer* yang berisi informasi berupa lima data blok terakhir dan lima data transaksi terakhir.

## B. Halaman Daftar Blok

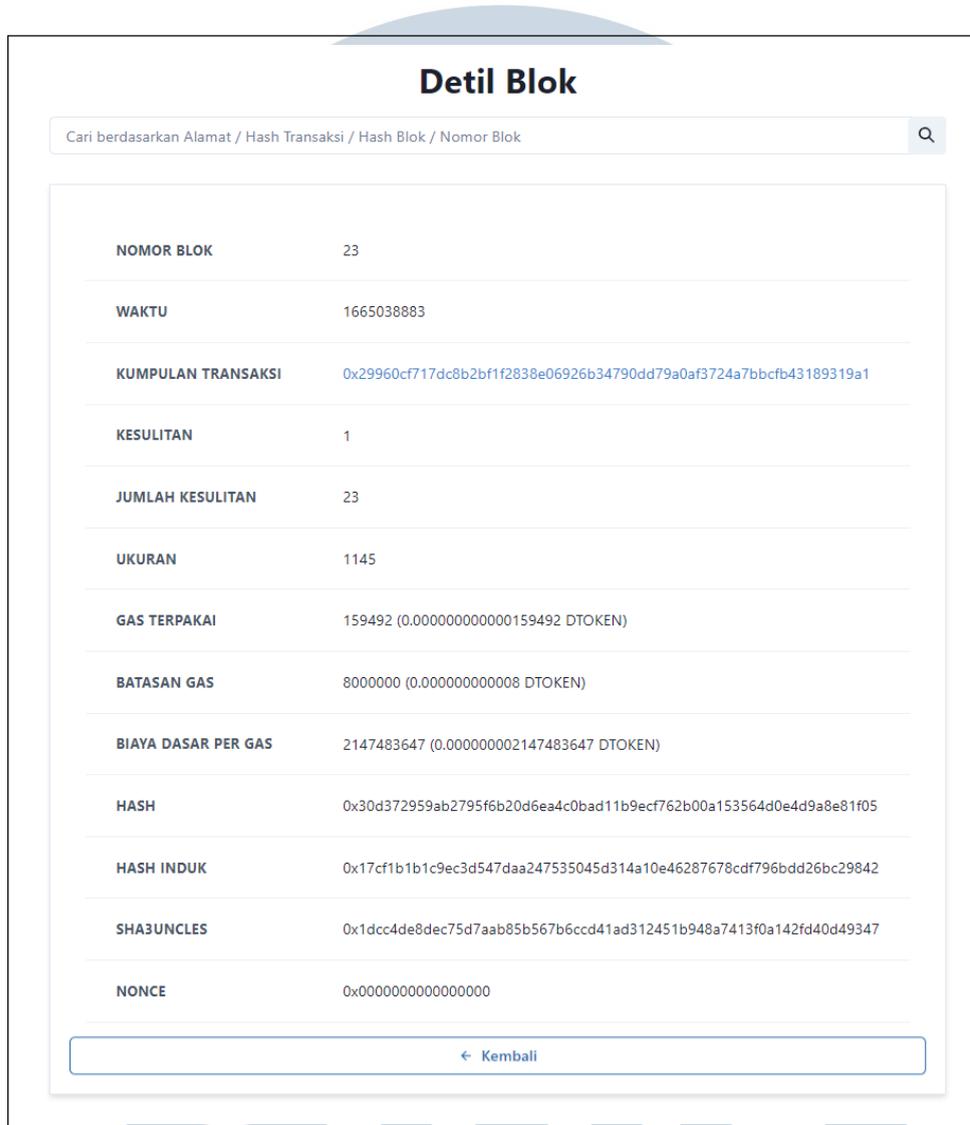


BLOK	UMUR / WAKTU	JUMLAH TRANSAKSI	HASH	GAS TERPAKAI	BATASAN GAS
544	1670307818	1	0x9074a29e1947621529377557a7c185a451103b1356a4c5209119a485a7	0.000000000000091937 DTKFN	0.0000000000000000 DTKFN
343	1670307762	1	0x90b0f347b3908a298c1acc8edc211a0bc731976056b8c070d4060384857234	0.0000000000000021 DTKFN	0.0000000000000000 DTKFN
542	166811279	1	0x77341d4f1608ba75074716a4a445e14b788034f56a764859a8b6a6a871c	0.000000000000007437 DTKFN	0.0000000000000000 DTKFN
541	1669707197	1	0x53034a18180a4533758a491a7a68034975a007036615a1970407019	0.000000000000125507 DTKFN	0.0000000000000000 DTKFN
340	1669698990	1	0x54acc00d01162bbd1cd1b008d0e0c1b0bd528912b142a3616d07062f	0.0000000000001235 DTKFN	0.0000000000000000 DTKFN
339	166963227	1	0x1e1718ac890c2d90c49d96cc099a2663f048c907b107130f96ab7142d1ca80	0.000000000000142692 DTKFN	0.0000000000000000 DTKFN
338	1669621197	1	0x730381373a7a6a6b39a4a1709ca82604b10a91e470087483a6a392a5122	0.000000000000040619 DTKFN	0.0000000000000000 DTKFN
337	1669621190	1	0x440f901b70811b8c230ca661d91f3b66bd2a49f052902d87103349ba224d2d	0.000000000000100875 DTKFN	0.0000000000000000 DTKFN
336	1669621058	1	0x2cd05c1ebceaa44a3b5c375a09a9b9f2a15d99ed300b9d9f34b336068b61	0.000000000000123048 DTKFN	0.0000000000000000 DTKFN
335	1669621034	1	0x089d055aadb106b509bca906a9311b45ee147165c58bd4a7968a10a50b8ed	0.00000000000014767 DTKFN	0.0000000000000000 DTKFN

Gambar 3.22. Halaman daftar blok pada situs web *blockchain explorer*

Pada Gambar 3.22 merupakan tampilan halaman daftar blok dari situs web *blockchain explorer* yang berisi informasi mengenai daftar data blok yang terdapat pada *blockchain*.

### C. Halaman Detail Blok



Detail Blok	
NOMOR BLOK	23
WAKTU	1665038883
KUMPULAN TRANSAKSI	0x29960cf717dc8b2bf1f2838e06926b34790dd79a0af3724a7bbcfb43189319a1
KESULITAN	1
JUMLAH KESULITAN	23
UKURAN	1145
GAS TERPAKAI	159492 (0.000000000000159492 DTOKEN)
BATASAN GAS	8000000 (0.0000000000008 DTOKEN)
BIAYA DASAR PER GAS	2147483647 (0.000000002147483647 DTOKEN)
HASH	0x30d372959ab2795f6b20d6ea4c0bad11b9ec762b00a153564d0e4d9a8e81f05
HASH INDUK	0x17cf1b1b1c9ec3d547daa247535045d314a10e46287678cdf796bdd26bc29842
SHA3UNCLES	0x1dcc4de8dec75d7aab85b567b6ccdd41ad312451b948a7413f0a142fd40d49347
NONCE	0x0000000000000000

Gambar 3.23. Halaman detail blok pada situs web *blockchain explorer*

Pada Gambar 3.23 merupakan tampilan halaman detail blok dari situs web *blockchain explorer* yang berisi informasi detail dari suatu blok pada *blockchain*.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

#### D. Halaman Daftar Transaksi

HASH TRANSAKSI	BLOK	ASAL	TUJUAN	NILAI
0x02956211961a6a081d1e5360a6c6681aeb18107408660559a9e4476358	344	0x1c81660540w7C069192D3F715d0B6485314981	0xC8D31AD4D063C3895564762w4F6d1190DC1c3f	0 DTOKEN
0x4142e04123e61229e322bcb9e99e99e21235d6dacc4bbabbede10c0c794145	343	0x4f5203771190769Ace7D1004f939894fC299c2	0x02066040e2c099192D3L715aD0e64853614981	2 DTOKEN
0x091537a89167e407445d68684d98a6a7a069670c4f5d8117075758d	340	0x4f5203771190769Ace7D1004f939894fC299c2	0xC8D31AD4D063C3895564762w4F6d1190DC1c3f	0 DTOKEN
0x8a4e45e6a81076d91925871f4a6876315e0a050e45015a1618791848	341	0x17A3489458771a7b3786a0551d91663685d5	0xC8D31AD4D063C3895564762w4F6d1190DC1c3f	0 DTOKEN
0x4f9fbc6c121c2b108f110ab738f080c0f100004c300263842c32b271220	340	0x4f5203771190769Ace7D1004f939894fC299c2	0x1238422006c2902f311903500111390898d54	0 DTOKEN
0x74019552962f10bc96dc77d52d57a78d5210400e6d5464104b0c0f6923a	339	0x4f5203771190769Ace7D1004f939894fC299c2	0xC8D31AD4D063C3895564762w4F6d1190DC1c3f	0 DTOKEN
0x8d1f0193e20e918f469e945b73856206e6068d4e488271aee0a020911e	338	0x4f5203771190769Ace7D1004f939894fC299c2	0xF97b766A9110C4274F9003244199f88wC08100	0 DTOKEN
0x072a1966a30904c9b2228f1aad9d2286c0b25216c91a616141c76a480e41	337	0x4f5203771190769Ace7D1004f939894fC299c2	0x97b766A9110C4274F9003244199f88wC08100	0 DTOKEN
0x7e0af723b240474073ee5ea175a6224359e95963e2d706c9608d5d61	336	0x19A3489575827d67b3786c0554D59636852d5	0x108C13435e4763bcC19C48385D80116877d91c8D	0 DTOKEN
0xe219195e6e39200d1Bc9b4836587d6d7ce0d6ec650aeb8d7a706338838	335	0x4f5203771190769Ace7D1004f939894fC299c2	0x108C13435e4763bcC19C48385D80116877d91c8D	0 DTOKEN

Gambar 3.24. Halaman daftar transaksi pada situs web *blockchain explorer*

Pada Gambar 3.24 merupakan tampilan halaman daftar transaksi dari situs web *blockchain explorer* yang berisi informasi mengenai daftar data transaksi yang terdapat pada *blockchain*.





## F. Halaman Detail Alamat

**Detail Alamat**

**RANGKUMAN**

Alamat:  
0x4f5203777190769Ace7D1804f859b0c4fC299c2  
Saldo:  
79.832109355 CTOKEN

Cari berdasarkan Alamat / Hash Transaksi / Hash Blok / Nomor Blok

HASH TRANSAKSI	BLOK	ASAL	TUJUAN	NILAI
0x34142ac412346132de32ebc99e969e2423d5ddace94baabede10cfe794745	343	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xbCB66D40e2C069192D3E715aD8e4B53614981	2 DTOKEN
0x0913326b9b52a0443c05404d9ac67aa959627cdd5af91412e73759fd	342	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xC3D31404D853C38595564262aF6ed1592DC1c3f	0 DTOKEN
0x470fbc6d312752b18b71aab738708c0af7e880947c30a3ba42ac3ab271226	340	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0x1228422D8742803fa1790a5B6d171a98B8B6E4	0 DTOKEN
0x374a1953963D10bc96de3774526d7a78e531e400e6dad647e4b0cc8f6923a	339	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xC3D31404D853C38595564262aF6ed1592DC1c3f	0 DTOKEN
0xbcd9f013f3e20e91fbab9c94567385620eab0606b4e4e827baee0d0911e	338	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xE97bb766A9110CA27eE9003244199E8Bac08100	0 DTOKEN
0x07aa1966a30904c9bd2289f1aae9a2b9f5cb5216c91a616741c176a490a41	337	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xE97bb766A9110CA27eE9003244199E8Bac08100	0 DTOKEN
0xe2191995c6e439200d18c99a3436587a0870e60e6c650a8a67a70638838	335	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0x108C13435e4763bec19C4B385D80116877a91c8D	0 DTOKEN
0x27a46942269a1cc0beb183b5d88bd54eaa79e6d8b9495d8c78745559e5af	333	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0x108C13435e4763bec19C4B385D80116877a91c8D	0 DTOKEN
0x28555f5b959454dbd88fa934355a617b0D4a7b1300c8b718e311aa7d47	332	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xE97bb766A9110CA27eE9003244199E8Bac08100	0 DTOKEN
0x2450672614633ec145cc36595ae9ff8ac629ae02hd43af654962a69046	331	0x4f5203777190769Ace7D1804f859b0c4fC299c2	0xE97bb766A9110CA27eE9003244199E8Bac08100	0 DTOKEN

Tampilkan 10 Data

Halaman 1 dari 25

Gambar 3.26. Halaman detail alamat pada situs web *blockchain explorer*

Pada Gambar 3.26 merupakan tampilan halaman detail alamat dari situs web *blockchain explorer* yang berisi informasi detail dari suatu alamat beserta daftar transaksi yang telah dilakukan oleh alamat tersebut pada *blockchain*.

### 3.4 Kendala dan Solusi yang Ditemukan

Pada pelaksanaan kerja magang ini, terdapat beberapa kendala yang ditemukan beserta solusi atau hal yang dilakukan penulis untuk mengatasi kendala tersebut, yaitu:

- Penulis mengalami kesulitan dalam memahami *system requirement* yang telah ditentukan karena kurangnya pengetahuan dan pengalaman mengenai Web3 dan teknologi *blockchain*. Solusinya adalah dengan mempelajari lebih lanjut tentang Web3 dan teknologi *blockchain*, baik melalui membaca artikel atau menonton video, serta mengikuti pelatihan tentang *blockchain* yang telah disediakan oleh tempat magang.
- Waktu yang dibutuhkan untuk memuat situs web *blockchain explorer* cukup lama, terutama ketika menampilkan daftar blok dan daftar transaksi. Solusi

untuk masalah ini yaitu dengan menerapkan fitur *pagination* pada *frontend* dan juga membuat fitur yang memungkinkan *frontend* untuk memilih jumlah data dan kolom yang diinginkan saat memanggil API. Dengan demikian, *frontend* dapat memilih sendiri jumlah data dan kolom yang diperlukan untuk ditampilkan.

- Kurangnya dokumentasi dan referensi mengenai beberapa teknologi *blockchain* yang digunakan dalam pengembangan sistem *blockchain* ini. Solusinya yaitu dengan berdiskusi atau bertanya dengan mentor atau tim *developer* yang sudah mendalami teknologi tersebut.

