**PERANCANGAN LEDGER PADA APLIKASI PENCATATAN DAN TRACKING DOKUMEN DENGAN IMPLEMENTASI BLOCKCHAIN**

***(LEDGER DESIGN ON DOCUMENT RECORDING AND TRACKING APPLICATION WITH BLOCKCHAIN IMPLEMENTATION)***

**TUGAS AKHIR**

Disusun sebagai syarat mata kuliah Tugas Akhir

Program Studi S1 Teknik Komputer

Disusun oleh:

**RIZAL NUR FAIZI**

**1103184114**

****

**FAKULTAS TEKNIK ELEKTRO**

**UNIVERSITAS TELKOM**

**BANDUNG**

**2017**

# LEMBAR PENGESAHAN

**TUGAS AKHIR**

**PERANCANGAN LEDGER PADA APLIKASI PENCATATAN DAN TRACKING DOKUMEN DENGAN IMPLEMENTASI BLOCKCHAIN**

***(LEDGER DESIGN ON DOCUMENT RECORDING AND TRACKING APPLICATION WITH BLOCKCHAIN IMPLEMENTATION)***

**Telah disetujui dan disahkan sebagai Buku Tugas Akhir**

**Program Studi Teknik Komputer**

**Fakultas Teknik Elektro**

**Universitas Telkom**

**Disusun oleh:**

**RIZAL NUR FAIZI**

**1103184114**

**Bandung, 23 Juli 2022**

|  |  |
| --- | --- |
| Pembimbing I  **Dr. Yudha Purwanto, S.T., M.T.**  NIP. 02770066 | Pembimbing II  **Muhammad Faris Ruriawan, S.T., M.T.**  NIP. 20920031 |

# ABSTRAK

Pada saat ini kebanyakan orang sudah mulai beralih dari dokumen secara fisik ke dokumen secara digital. Dokumen secara fisik mungkin hanya dibutuhkan ketika sedang mengurus hal-hal yang bersifat administratif dan membutuhkan copy dari dokumen tersebut sebagai bukti fisik. Pada saat ini dokumen digital banyak diimplementasikan pada berbagai macam hal seperti sertifikat, piagam penghargaan, dan bahkan ijazah. Namun hal tersebut rawan akan pemalsuan dan jika hal tersebut dilakukan pendataan, akan sulit untuk dideteksi bila pada database ada yang merubah data tersebut atau informasi mengenai data tersebut karena kurangnya transparansi.

Maka dari itu, blockchain bisa menjadi salah satu sarana atau jalan keluar dalam mengatasi hal tersebut. Karena blockchain unggul dalam hal transparansi data sehingga data tersebut akan terjaga validitasnya. Selain itu karena blockchain hanya menggunakan sistem read and write pada setiap transaksinya maka transaksi-transaksi yang sudah dilakukan akan tetap ada dan tidak berubah.

Maka dari itu diharapkan dari Tugas Akhir ini, dapat terwujudnya suatu website yang dapat berguna untuk menyimpan dokumen – dokumen digital dengan memanfaatkan teknologi blockchain untuk metode penyimpanannya. Selain dapat menyimpan juga bisa digunakan untuk melakukan tracking suatu dokumen yang sebelumnya telah disimpan.

**Kata Kunci:** *Blockchain, Dokumen Digital, Tracking*

# DAFTAR ISI

[LEMBAR PENGESAHAN ii](#_Toc109505654)

[ABSTRAK iii](#_Toc109505655)

[DAFTAR ISI iv](#_Toc109505656)

[DAFTAR GAMBAR v](#_Toc109505657)

[DAFTAR TABEL vi](#_Toc109505658)

[BAB I PENDAHULUAN 1](#_Toc109505659)

[1.1. Latar Belakang Masalah 1](#_Toc109505660)

[1.2. Rumusan Masalah 2](#_Toc109505661)

[1.3. Tujuan dan Manfaat 3](#_Toc109505662)

[1.4. Batasan Masalah 3](#_Toc109505663)

[1.5. Metode Penelitian 4](#_Toc109505664)

[1.6. Jadwal Pelaksanaan 4](#_Toc109505665)

[BAB II TINJAUAN PUSTAKA 5](#_Toc109505666)

[2.1. Penggunaan Bahasa Indonesia yang Baik dan Benar 5](#_Toc109505667)

[2.2. Format Penulisan Tugas Akhir 7](#_Toc109505668)

[**2.2.1** **Kertas** 7](#_Toc109505669)

[**2.2.2.** **Pengetikan** 7](#_Toc109505670)

[**2.2.3.** **Penomoran Halaman** 7](#_Toc109505671)

[**2.2.4.** **Ketentuan Halaman Sampul** 8](#_Toc109505672)

[2.3. Ketentuan Penggunaan Gambar, Tabel dan Persamaan 8](#_Toc109505673)

[**2.3.1.** **Penyisipan Gambar** 8](#_Toc109505674)

[**2.3.2.** **Penyisipan Tabel** 9](#_Toc109505675)

[**2.3.3.** **Penulisan Rumus atau Persamaan** 10](#_Toc109505676)

[2.4. Penulisan Kutipan format IEEE. 10](#_Toc109505677)

[**2.4.1.** **Kutipan Langsung** 10](#_Toc109505678)

[**2.4.2.** **Kutipan Tidak Langsung** 11](#_Toc109505679)

[**2.5.** **Penggunaan Referensi** 11](#_Toc109505680)

[**2.6.** **Penggunaan *Footnote*** 11](#_Toc109505681)

[**2.7.** **Format Penulisan Daftar Sumber Pustaka** 19](#_Toc109505682)

[**2.7.1.** **Buku** 19](#_Toc109505683)

[**2.7.2.** **Artikel dari proceeding (seminar, workshop)** 20](#_Toc109505684)

[**2.7.3.** **Sumber *online* (seperti wikipedia)** 20](#_Toc109505685)

[BAB III PERANCANGAN SISTEM 21](#_Toc109505686)

[3.1. Desain Sistem 21](#_Toc109505687)

[Dalam bab ini diuraikan secara rinci cara dan pelaksanaan kerja, hasil pengamatan percobaan atau pengumpulan data dan informasi lapangan, pengolahan data dan informasinya. 21](#_Toc109505688)

[**3.1.1.** **Diagram Blok** 21](#_Toc109505689)

[**3.1.2.** **Fungsi dan Fitur** 21](#_Toc109505690)

[**3.2.** **Desain Perangkat Keras** 21](#_Toc109505691)

[**3.2.1.** **Spesifikasi Komponen** 21](#_Toc109505692)

[**3.3.** **Desain Perangkat Lunak** 21](#_Toc109505693)

[**3.3.1.** **Spesifikasi Sub Sistem** 21](#_Toc109505694)

[BAB IV HASIL DAN ANALISIS 21](#_Toc109505695)

[4.1. Hasil Percobaan 21](#_Toc109505696)

[**4.1.1.** **Pengujian Parameter A** 22](#_Toc109505697)

[**4.1.2.** **Pengujian Parameter B** 22](#_Toc109505698)

[**4.2.** **Analisis** 22](#_Toc109505699)

[**4.2.1.** **Analisis Hubungan Parameter A terhadap Tujuan A** 22](#_Toc109505700)

[**4.2.2.** **Analisis Hubungan Parameter B terhadap Tujuan A** 22](#_Toc109505701)

[**4.2.3.** **Analisis Hubungan Parameter A terhadap Tujuan B** 22](#_Toc109505702)

[**4.2.4.** **Analisis Hubungan Parameter A terhadap Tujuan B** 22](#_Toc109505703)

[BAB V SIMPULAN DAN SARAN 22](#_Toc109505704)

[5.1. Simpulan 22](#_Toc109505705)

[5.2. Saran 22](#_Toc109505706)

[DAFTAR PUSTAKA 24](#_Toc109505707)

[LAMPIRAN 25](#_Toc109505708)

# DAFTAR GAMBAR

[**Gambar 2. 1** Perbandingan sentralisasi, desentralisasi dan distribusi 6](#_Toc109573586)

[**Gambar 2. 2** Daftar riwayat transaksi 6](#_Toc109573587)

[**Gambar 3. 1** Gambaran Umum Desain Sistem Ledger 11](#_Toc109573609)

[**Gambar 3. 2** Diagram Blok 12](#_Toc109573610)

[**Gambar 3. 3** Data flow diagram level 0 14](#_Toc109573611)

[**Gambar 3. 4** Data flow diagram level 1 15](#_Toc109573612)

[**Gambar 3. 5** Data flow diagram level 2 proses 1 16](#_Toc109573613)

[**Gambar 3. 6** Data flow diagram level 2 proses 2 17](#_Toc109573614)

[**Gambar 3. 7** Data flow diagram level 2 proses 3 17](#_Toc109573615)

[**Gambar 3. 8** Diagram alir mengirim file 18](#_Toc109573616)

[**Gambar 3. 9** Diagram alir mengirim file 19](#_Toc109573617)

# DAFTAR TABEL

**Tabel 1.1** Contoh Jadwal dan *Milestone*

**Tabel 1.1** Hubungan Antara *Input* dan *Output*

# BAB I PENDAHULUAN

## 1.1. Latar Belakang Masalah

Sekarang ini sistem pencatatan pada database masih bersifat terpusat atau centralized, sehingga hanya dapat diakses dan dirubah oleh administrator database. Database yang dipegang oleh otoritas terpusat biasanya menyimpan banyak salinan data untuk keamanan pengelolaan server data yang tersimpan rentan mudah rusak. Selain itu, risiko lainnya dari database adalah data yang rentan dimodifikasi jika administrator lemah dari segi penjagaan keamanan. Sehingga ada rasa ketidakpercayaan oleh pemilik dokumen apabila ada seseorang yang merubah suatu isi dari dokumen tersebut.

Blockchain (distributed ledger) adalah database terdistribusi yang menyimpan catatan secara bertambah terus menerus. Blockchain bersifat decentralized sehingga dapat melindungi privasi, keterbukaan dan ketersediaan data [1]. Blockchain unggul dalam hal transparansi, sebab semua data valid yang sudah ditulis bisa dilihat dan diakses oleh publik. Blockchain akan mencatat data yang telah divalidasi dalam suatu blok, lalu dipadukan dalam rantai blok-blok yang terpadu. Blockchain hanya mencatatkan hasil transaksi yang valid, sifatnya tetap alias tidak bisa diubah [2]. Sehingga tidak akan terjadi manipulasi data karena setiap blok akan terhubung satu sama lain sehingga apabila terdapat data yang berbeda maka akan dapat dilihat maupun ditaracking. Untuk saat ini pemanfaatan implementasi teknologi blockchain masih belum terlalu masif dan hanya ada beberapa yang sudah menggunakan contohnya cryptocurrency.

Oleh karena permasalahan diatas penulis membuat tugas akhir ini dibutlah sistem untuk menanggulangi masalah diatas dengan menggunakan sistem pencatatan menggunakan blockchain. Sehingga dapat mencegah manilupasi data dan mudah untuk melakukan tracking sebuah dokumen karena setiap blok saling terhubung. Selain itu juga dalam segi keamanan penyimpanan pada blockchain jauh lebih baik dan juga adanya transparansi data.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan pada bagian sebelumnya, maka rumusan masalah dalam pengerjaan Tugas Akhir ini adalah sebagai berikut :

1. Bagaimana proses pembuatan ledger pada website berbasis blockchain ?
2. Bagaimana mengimplementasikan blockchain pada website pencatatan dan tracking dokumen digital berbasis blockchain ?

## 1.3. Tujuan dan Manfaat

Dari pengerjaan tugas akhir ini, diharapkan tercapainya tujuan seperti berikut ini:

1. Mengetahui cara pembuatan ledger pada website berbasis blockchain.
2. Mengetahui implementasi blockchain pada website pencatatan dan tracking dokumen digital berbasis blockchain.

## 1.4. Batasan Masalah

Batasan masalah yang terdapat pada Tugas Akhir kali ini adalah sebagai berikut ini :

1. Dalam pengerjaan blockchain nya akan menggunakan identity protocol dari ceramic idx. Sedangkan ledger yang akan digunakan adalah ethereum
2. Jenis data yang akan disimpan adalah berupa *buffer* file yang akan dikirim menuju.
3. Nantinya data yang tersimpan harus bisa di tracking melalui riwayat transaksi.

## 1.5. Metode Penelitian

Metode penelitian yang akan digunakan penulis pada tugas akhir kali ini adalah sebagai berikut :

1. Studi Literatur

Studi literatur yang dilakukan adalah dengan mengumpulkan dan mempelajari tentang konsep dan teori dari blockchain dan algoritma pemrograman dari buku, artikel, jurnal, E-Book, sebagai dasar teori yang akan dikaji ulang sebagai bahan untuk pembuatan tugas akhir.

1. Pengumpulan Data

Pengambilan data dilakukan dengan melakukan input data, lalu data yang diperoleh akan disimpan kedalam Blockchain. Dari data yang tersimpan tersebut akan dapat dilakukan tracking terhadap setiap perubahan data yang dilakukan.

1. Perancangan Aplikasi

Dalam perancangan aplikasi, akan dibagi menjadi beberapa bagian yaitu front-end, back-end, dan ledger(blockchain). Adapun aplikasi yang akan dibuat adalah Aplikasi Pencatatan dan Tracking berbasis blockchain.

1. Pengujian Aplikasi

Pada tahapan ini akan dilakukan sebuah pengujian terhadap aplikasi yang sudah kita buat. Pada tahap ini kita akan mengetahui hasil implementasi blockchain pada sebuah aplikasi pencatatan. Disinilah kita akan memperbaiki bila mana ada yang masih salah.

## 1.6. Jadwal Pelaksanaan

Berikut adalah jadwal pelaksanakan yang diharapkan bisa menjadi acuan untuk mengerjakan tugas akhir :

**Tabel 1.1** Jadwal dan *Milestone*.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Deskripsi Tahapan** | **Durasi** | **Tanggal Selesai** | ***Milestone*** |
| 1 | Desain Sistem | 1 minggu | 3 Feb 2022 | Diagram Blok serta mengumpulkan data – data yang akan digunakanm |
| 2 | Pemilihan dan instalasi Komponen | 2 minggu | 24 Feb 2022 | Membuat list komponen yang akan digunakan dan melakukan instalasi |
| 3 | Pembuatan dan pembangunan sistem | 2 bulan | 21 Apr 2022 | Menyelesaikan seluruh pembuatan sistem blockchain |
| 4 | Pengujian sistem dan memperbaiki bug | 1 bulan | 19 Mei 2022 | Melakukan pengujian aplikasi agar dapat berjalan dengan baik |
| 5 | Penyusunan laporan/buku TA | 2 bulan | 14 Juli 2022 | Buku TA selesai |

# BAB II TINJAUAN PUSTAKA

## 2.1. Blockchain

Secara singkat blockchain merupakan basis data terdistribusi dari record atau public ledger dari semua transaksi atau peristiwa digital yang sudah dieksekusi dan lalu dibagikan pada pihak yang berpartisipasi. Setiap transaksi yang dilakukan diverifikasi dengan consensus mayoritas oleh partisipan yang ada pada sistem. Setelah dimasukan ke dalam sistem, data tidak akan pernah bisa dihapus. Transaksi yang dilakukan akan direkam dalam satu buah block, dimana setiap block ini terhubung dengan block lain yang ada sebelum atau sesudah sebuah transaksi dilakukan.

Teknologi buku besar terdistribusi atau Distributed ledger technologies (DLT) termasuk blockchain, menggabungkan penggunaan kriptografi dan jaringan terdistribusi untuk mencapai bentuk baru dari pembuatan dan penyimpanan catatan yang dirancang untuk ketahanan terhadap kerusakan dan kekekalan [3]. Terdapat 3 pilar dalam blockchain yaitu adalah Decentralization, Transparency, Immutability.

### Desentralisasi

Sistem desentralisasi pada blockchain adalah sistem yang tidak memiliki otoritas sentral, tetapi masih bisa bekerja sama baiknya seperti seakan-akan ada otoritas sentral di belakangnya. Desentralisasi blockchain juga dapat diartikan sebagai suatu sistem yang keseluruhan pengambilan keputusannya diserahkan kepada para pengguna sistem tersebut. Dalam sistem desentralisasi, informasi tidak disimpan oleh satu entitas tunggal. Faktanya, semua pengguna yang berada di jaringan memiliki segala informasi dan jika ingin berinteraksi dapat melakukannya secara langsung tanpa melalui pihak ketiga [4].

Chart, radar chart

Description automatically generated

**Gambar 2. 1** Perbandingan sentralisasi, desentralisasi dan distribusi

### Transparansi

Salah satu konsep yang paling menarik dan disalahpahami dalam teknologi blockchain adalah “transparansi”. Beberapa mengatakan bahwa blockchain memberi Anda privasi sementara beberapa orang mengatakan itu transparan. Dalam blockchain identitas seseorang disembunyikan melalui kriptografi yang kompleks dan hanya diwakili oleh alamat publik mereka [4]. Jadi setiap transaksi atau pertukaran data akan tercatat dan tetap bisa dicari riwayatnya. Jadi, sementara identitas asli orang tersebut aman, Anda masih akan melihat semua transaksi yang dilakukan oleh alamat publik mereka.

Graphical user interface, application

Description automatically generated

**Gambar 2. 2** Daftar riwayat transaksi

### Immutability

*Immutability* atau kekekalan dalam konteks blockchain, berarti bahwa sekali sesuatu telah dimasukkan ke dalam *blockchain*, itu tidak dapat dirusak ataupun dimanipulasi. Alasan tersebut ada karena dalam *blockchain* menggunakan fungsi hash kriptografi untuk mengamankan datanya. hashing berarti mengambil string input dengan panjang berapa pun dan memberikan output dengan panjang tetap. Dalam konteks cryptocurrency seperti bitcoin, transaksi diambil sebagai input dan dijalankan melalui algoritma hashing (bitcoin menggunakan SHA-256) yang memberikan output dengan panjang tetap.

**Tabel 2. 1** Hashing menggunakan SHA-256

|  |  |
| --- | --- |
| **Input** | **Hash** |
| Hallo | 753692ec36adb4c794c973945eb2a99c1649703ea6f76bf259abb4fb838e013e |
| Saya rizal | d8ce85518016724843df8165f846a0492cab5f27f5b32a393d865bde48d95c7b |

## 2.2. Ethereum

Ethereum merupakan sebuah teknologi yang ada pada blockchain yang menyediakan platform terdesentralisasi untuk membangun aplikasi blockchain terintegrasi lengkap dengan operasi smart contract [5]. Ethereum dapat digunakan untuk membuat smart contract yang dapat dieksekusi dalam aplikasi terdesentralisasi menggunakan blockchain, bahkan sekaligus dapat membuat aturan semaunya untuk kepemilikan sendiri, format transaksi, dan fungsi-fungsi transisinya [2].

## 2.3. Polygon

Polygon mumbai adalah *testnet* (jaringan eksperimental) dari Jaringan Polygon yang digunakan untuk menguji, membuat, bahkan memodifikasi fungsi serta memantau kinerja jaringan blockchain. Jaringan Polygon adalah solusi blockchain lapisan kedua (layer 2) yang dibangun di atas jaringan Ethereum, yang merupakan jaringan blockchain lapis 1. Jaringan lapis kedua adalah kerangka kerja atau protokol sekunder yang dibangun diatas sistem blockchain utama, atau dikenal sebagai lapis 1. Kehadiran protokol ini bertujuan untuk menjadi solusi atas leletnya kecepatan transaksi dan rendahnya skalabilitas transaksi yang dihadapi jaringan kripto.

## 2.4. Smart Contract

Smart contract pertama kali digagas pada tahun 1994 oleh seorang pakar kritpografi, ilmuwan komputer, dan juga sarjana hukum yaitu Nick Szabo dengan tujuan dibuatnya adalah untuk membantu kesepakatan antara dua belah pihak dalam proses pertukaran uang, barang, saham atau apapun yang bernilai secara transparan, dan bebas konflik dengan menghindari pihak ke 3 saat melakukan transaksi [6].

**2.5. Brownie**

Brownie adalah *tools* yang digunakan untuk *development* dan pengujian *smart contract* yang berbasis python. Banyak sekali framework yang dapat dijalankan untuk mempermudah proses pengembangan dan pengujian kontrak sebelum dijalankan pada jaringan *blockchain.* Karena brownie dapat menjalankan kontrak pada blockchain lokal. Sehingga nantinya pada saat proses produksi dapat meminimalisir eror karena sudah diuji terlebih dahulu pada jaringan lokal. Setelah di jaringan lokal aman maka akan dilakukan migrasu

**2.6. Alchemy**

*Footnote* atau catatan kaki adalah keterangan yang dicantumkan pada margin bawah halaman buku (biasanya dicetak dengan huruf lebih kecil daripada huruf di dalam teks yang berfungsi un

**2.7. Penelitian Terkait**

Daftar Pustaka mengikuti format IEEE. Berikut ini dijelaskan beberapa tipe penulisan pustaka.

**2.7.1. Buku**

Format standar:

[#] A.A. Penulis/editor, *Judul: Subjudul*, Edisi (jika bukan yang pertama), Vol. (jika dibuat beberapa volume). Tempat publikasi: *Publisher*, Tahun, halaman (jika diperlukan).

Contoh:

* Jika satu penulis:

[20] [R. R. Yager](http://www.sciencedirect.com/science/article/pii/S0020737377800084" \l "!), *Multiple objective decision-making using fuzzy sets*, International Journal of Man-Machine Studies, Vol. 9, No. 4, pp.375-382, Jul. 1977.

* Lebih dari satu penulis:

[11] R. Hayes, G. Pisano, D. Upton, and S. Wheelwright, Operations, Strategy, and Technology: Pursuing the competitive edge. Hoboken, NJ : Wiley, 2005.

* Penulis tiga orang atau lebih. Jika tidak lebih dari enam orang, maka semua nama penulis harus disebutkan. Jika lebih dari enam, bisa digunakan “et al.” setelah nama penulis pertama.

[7] R. Hayes, G. Pisano, D. Upton, and S. Wheelwright, Operations, Strategy, and Technology: Pursuing the competitive edge. Hoboken, NJ : Wiley, 2005.

* Buku berseri:

[21] M. Bell, et al., Universities Online: A survey of online education and services in Australia, Occasional Paper Series 02-A. Canberra: Department of Education, Science and Training, 2002.

* Penulis Lembaga:

[32] World Bank, Information and Communication Technologies: A World Bank group strategy. Washington, DC : World Bank, 2002.

* Thesis / disertasi:

[43] H. Zhang, "Delay-insensitive networks," M.S. Thesis, University of Waterloo, Waterloo, ON, Canada, 1997.

**2.7.2. Artikel dari proceeding (seminar, workshop)**

Contoh untuk paper yang dipresentasikan dalam sebuah seminar:

[1] H. A. Nimr, "Defuzzification of the outputs of fuzzy controllers," presented at 5th International Conference on Fuzzy Systems, Cairo, Egypt, 1996.

**2.7.3. Sumber *online* (seperti wikipedia)**

Sebuah sumber elektronik atau *online* terkadang tidak disertai dengan informasi penulis/*publisher* yang jelas. Sehingga, informasi yang digunakan hanya URL dari sumber tulisan tersebut.

Sama seperti sumber publikasi lainnya, tanggal akses informasi *online* adalah suatu hal yang penting. Tanggal pengaksesan juga dicantumkan karena sumber *online* bisa berubah isinya antara pada saat diakses, dengan pada saat pembaca mengaksesnya

[1] J. Riley, "Call for new look at skilled migrants,"*The Australian*, p. 35, May 31, 2005. Available: Factiva, http://global.factiva.com. [Diakses 31 May 2005, 23:59:59 WIB].

# BAB III PERANCANGAN SISTEM

## 3.1. Desain Sistem

Dalam penelitian Tugas Akhir ini akan merancang sistem *ledger* untuk implementasi *blockchain* dan jaringan *peer-to-peer* untuk melakukan pengiriman data dokumen digital berbasis aplikasi web. Sehingga agar website dapat terhubung dengan jaringan blockchain dan jaringan peer-to-peer terdistribusi membutuhkan membutuhkan *smart contract* yang ditulis dalam bahasa pemrograman Solidity. Jaringan peer-to-peer terdistribusi yang digunakan yaitu InterPlanetary File System (IPFS) dan untuk jaringan blockchainnya menggunakan Polygon testnet. Dengan sistem Ledger seperti itu pengguna memiliki akses untuk dapat menyimpan, tracking, maupun mengirim dokumen digital dengan melakukan transaksi yang dikelola melalui metamask.

Graphical user interface, application, Teams

Description automatically generated

**Gambar 3. 1** Gambaran Umum Desain Sistem Ledger

### Diagram Blok

Berdasarkan rancangan yang sudah dibuat sebelumnya, alur kerja dari sistem ledger yang dibuat akan seperti pada diagram block dibawah ini :

Graphical user interface

Description automatically generated

**Gambar 3. 2** Diagram Blok

Seperti pada Gambar 3.2 diatas alur kerja dari sistem ledger ini adalah jika suatu user memberikan input maka akan diarahkan untuk terhubung dengan metamask. Melalui metamask user akan dihubungkan dengan smart contract agar nanti dapat terhubung dengan jaringan Polygon dan IPFS. Melalui smart contract seluruh transaksi dijalankan sesuai dengan kontrak yang sudah dibuat dan kemudian data diteruskan ke jaringan Polygon dan IPFS.

### Fungsi dan Fitur

Dibawah ini adalah penjelasan dari fitur – fitur yang terdapat pada diagram blok diatas.

1. User

User merupakan pihak yang melakukan input dokumen melalui web interface. User sendiri menggunakan alamat dari account metamask untuk sebagai autentikasi dari setiap pengguna. Jadi setiap user hanya perlu mempunyai metamask agar dapat menjalankan aplikasi ini.

1. Metamask

Metamask merupakan sebuah wallet crypto yang terdapat pada ekstensi pada browser. Metamask sendiri berfungsi untuk menghubungkan atau berinteraksi dengan dengan website berbasis blockchain.

1. Smart Contract

*Smart contract* berfungsi untuk menjalankan transaksi antara user dengan jaringan blockchain. Dengan *smart contract* pencatatan transaksi tidak akan melalui perantara pihak ketiga, sehingga seluruh aktifitas transaksi tidak bersifat sentralisasi atau terpusat.

1. Alchemy

Alchemy merupakan salah satu platform developer blockchain yang menyediakan API untuk dapat berinteraksi dengan jaringan blockchain. Selain itu alchemy dapat digunakan untuk monitoring dan tracking contract yang sedang berjalan.

1. Polygon

Jaringan Polygon adalah solusi blockchain layer 2 yang dibangun di atas jaringan Ethereum, yang merupakan jaringan blockchain layer 1. jaringan polygon hadir untuk mengatasi hambatan transaksi di jaringan Ethereum, mulai dari masalah skalabilitas, mahalnya gass fee, hingga kecepatan transaksi.

1. IPFS

IntraPlanetary File System atau IPFS adalah protokol dan jaringan peer-to-peer untuk menyimpan dan berbagi data dalam sistem file terdistribusi. Setiap konten yang menggunakan protokol IPFS akan dilengkapi dengan content identifier atau CID sebagai hash nya. IPFS juga sangat kompatibel dengan semua format file digital yang ada saat ini, mulai dari gambar, stream video, dan database.

## 3.2. Perancangan Sistem Ledger

Dalam melakukan perancangan sistem ledger agar dapat terhubung dengan *back-end* dan *front-end* membutuhkan beberapa *tools* dan *library* yang digunakan antara lain yaitu Solidity, IPFS, Polygon, dan Alchemy. *Smart Contract* akan menjalankan beberapa *function* yang sudah dibuatantara lain *setListingFee* (gas fee), *uploadFile* (mengirim data), *getUserFiles* (kepemilikan data). *setListingFee* merupakan fungsi untuk menetapkan *gas fee* yang dibutuhkan pada transaksi yang akan dijalankan. *uploadFile* merupakan fungsi untuk untuk mengirimkan data ke IPFS sekaligus juga menuliskan hash ke blockchain agar tercatat pada *node*. *getUserFiles* merupakan fungsi untuk menampilkan data yang sudah sudah dimasukkan ke IPFS berdasarkan hash pada address user yang digunakan pada saat upload file.

### 3.2.1. Diagram Alir Data

Untuk memudahkan dalam memahami aliran data pada ledger maka dibuatkan *Data Flow Diagram* atau diagram alir data sebagai berikut :

* **DFD Level 0**

Pada DFD level 0 terdapat dua entitas eksternal yaitu User dan Alchemy. User sendiri merupakan pihak yang melakukan input data berupa file dokumen yang akan disimpan. Sedangkan Alchemy merupakan API provider yang berfungsi untuk menghubungkan ke jaringan *blockchain* dan *IPFS.* Selain itu juga disini terdapat satu proses yaitu web interface yang berfungsi untuk menjalankan proses pengiriman maupun pembacaan data.

Diagram

Description automatically generated with medium confidence

**Gambar 3. 3** Data flow diagram level 0

* **DFD Level 1**

Pada DFD level 1 terdapat tiga proses yang berjalan yaitu mengirim data, menuliskan data dan membaca data. Pada proses mengirim data entitas user akan melakukan input file yang kemudian mengeluarkan *output* untuk melakukan konfirmasi. Setelah dikonfirmasi file akan diubah menjadi *buffer* dan *hash* kemudian diteruskan menuju alchemy. Setelah pada alchemy data yang masuk akan ditulis menuju IPFS dan Blockchain dengan smart contract, pada proses ini akan meghasilkan *output* *gas fee* yang harus dibayar. Setelah *gas fee* dibayar maka data akan ditulis dan selanjutnya akan menjalankan fungsi membaca data. Pada fungsi membaca data akan menghasilkan *output CID file* dan *TxHash* yang akan diteruskan menuju user untuk dapat melihat file yang berhasil dikirim.

Graphical user interface, application, Teams

Description automatically generated

**Gambar 3. 4** Data flow diagram level 1

* **DFD Level 2**

Pada DFD level 2 merupakan penjelasan lebih terinci pada setiap proses yang ada pada level 1. Disini ada 3 proses yang dibuat yaitu proses 1 : mengirim data, proses 2 : menuliskan data dan proses 3 : membaca data. Untuk diagram dari setiap prosesnya adalah sebagai berikut ini :

1. **DFD Level 2 Proses 1**

Pada proses 1 user akan melakukan input file kemudian mendapat konfirmasi untuk melakukan upload. Setelah mendapat konfirmasi akan masuk ke proses membaca data, disini akan menganbil data yang berupa nama file, ukuran file, waktu upload file, dan data file. Setelah itu data akan masuk pada proses mengubah file. Disini file akan diambil buffer dan hashnya untuk nanti diteruskan ke proses terakhir. Yang terakhir ada proses mengirim data file yang berupa buffer dan hash untuk diteruskan menuju alchemy.

Graphical user interface, application

Description automatically generated with medium confidence

**Gambar 3. 5** Data flow diagram level 2 proses 1

1. **DFD Level 2 Proses 2**

Pada proses 2 user akan mendapatkan *gas fee* yang menggunakan token polygon mumbai yang harus dibayar agar proses transaksi berjalan. Setelah *gas fee* dibayar maka transaksi akan dieksekusi oleh *smart contract,* disini data buffer yang sudah dikirim pada alchemy akan ditulis menuju IPFS dan hash dari informasi file akan dituliskan ke blockchain.

Graphical user interface

Description automatically generated

**Gambar 3. 6** Data flow diagram level 2 proses 2

1. **DFD Level 2 Proses 3**

Pada proses 3 alchemy akan mengambil informasi user berupa *wallet address* untuk nantinya akan diambil data transaksi yang berupa *TxHash*. Didalam *TxHash* terdapat beberapa informasi mulai dari *CID file* untuk mengakses ke IPFS, data blok yang digunakan, dan semua yang tercatat pada saat transaksi. Jadi dengan *TxHash* user dapat melakukan tracking seluruh transaksi yang sudah dijalankan. *TxHash* akan tersimpan pada wallet yang digunakan oleh user, disini kita menggunakan *metamask* sebagai walletnya.

Diagram

Description automatically generated

**Gambar 3. 7** Data flow diagram level 2 proses 3

### 3.2.2. Diagram Alir

Agar dapat mengetahui alur dari sistem ledger yang akan dibuat, berikut adalah diagram alir pada sistem ledger yang dibagi menjadi 2 operasi utama yaitu mengirim file dan membaca file. Adapun diagram alir operasinya adalah sebagai berikut :

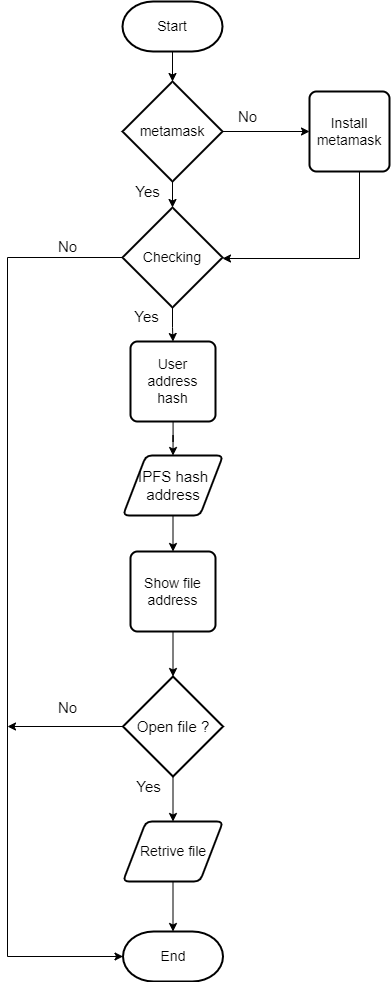
* **Mengirim File**

A picture containing diagram

Description automatically generated

**Gambar 3. 8** Diagram alir mengirim file

* **Membaca File**



**Gambar 3. 9** Diagram alir mengirim file

### 3.2.3. Pernacangan Smart Contract

Dalam perancangan smart contract disini dibuat *FileStorage.sol* sebagai smart contractnya menggunakan bahasa pemrograman Solidity. Dalam smart contract tersebut ada beberapa *function* seperti yang sudah dijelaskan sebelumnya yaitu ada *setListingFee, uploadFile, dan getUserFiles.* Berikut ini adalah penjelasan *function* dari smart contrat :

* ***setListingFee***

Merupakan fungsi untuk menjalankan operasi penetapan gas fee untuk setiap transaksi. Dalam operasinya fungsi ini dikombinasikan dengan *getMaticUsdPrice* yang memanfaatkan AggregatorV3Interface.sol dari contract chainlink agar bisa mendapatkan *price feed* (daftar harga terbaru). Dengan begitu penetapan gas fee akan mengikuti harga real time pada saat itu juga. Selain itu kelebihan menggunakan chainlink adalah dapat melakukan konversi dari network blockchain yang berbeda.

**Fungsi ListingFee**

**Deklarasi**

struktur **File** {

string **name**;

uint256 **size**;

string **uri**;

uint256 **uploadDate**;

}

**Algoritma**

1. constructor(address \_priceFeedAddress) {

priceFeedAddress = \_priceFeedAddress;

owner = payable(msg.sender);

}

1. modifier **onlyOwner**() {

require(msg.sender == owner, "only owner can use this");

}

**Implementasi**

1. Fungsi **setListingFee**(uint256 \_fee) public onlyOwner {

(uint256 price, uint256 decimals) = getMaticUsdPrice();

uploadFee = (\_fee \* 10\*\*decimals) / price;

}

1. Fungsi **getListingFee**() public view returns (uint256) {

return uploadFee;

}

**// mendapatkan harga matic/usd dari chainlink price feed**

1. Fungsi **getMaticUsdPrice**() internal returns (uint256, uint256) {

AggregatorV3Interface priceFeed =AggregatorV3Interface

(priceFeedAddress);

(, int256 price, , , ) = priceFeed.latestRoundData();

uint256 decimals = priceFeed.decimals();

return (uint256(price), decimals);

}

* ***uploadFile***

Merupakan fungsi untuk melakukan pendataan penyimpanan kedalam IPFS. Mekanisme operasi fungsi ini adalah akan mengubah file menjadi *buffer* yang nantinya akan masuk ke IPFS. Setiap data yang dimasukkan ke IPFS akan dikonfirmasi oleh *contract* melalui metamask dan apabila dikonfirmasi maka informasi file akan ditulis pada *blockchain.* Setiap transaksi yang berhasil akan mendapatkan *TxHash* (hash transaksi) sebagai bukti transaksi yang masuk ke dalam blockchain.

**Fungsi** upload\_file

**Deklarasi** (uploadFile)

**name** : string memory

**size** : uint256

**fileURI** : string memory

**Implementasi**

1. Set mapping(address => File[]) public usersFiles
2. Input msg.value == uploadFee
3. Output "Untuk upload file harus membayar gas fee"
4. Input newFile = File(name,size,fileURI, block.timestamp)
5. Output usersFiles[msg.sender].push(newFile)
6. Output FileAdded(msg.sender, name)
7. Write owner.transfer(msg.value)

* ***getUserFiles***

Merupakan fungsi untuk melakukan pembacaan kepemilikan file berdasarkan address user. Dalam pemanggilan fungsi ini hanya akan melakukan pembacaan data tanpa adanya perubahan data yang dilakukan.

**Fungsi** getUserFiles

**Implementasi**

1. getUserFiles(address \_user) public view returns (File[] memory)
2. require(msg.sender == \_user, "only user can check his own files")
3. return usersFiles[\_user]

### 3.2.4 Spesifikasi Komponen

Dalam pengerjaan website aplikasi blockchain, spesifikasi laptop yang digunakan adalah sebagai berikut :

1. CPU : Intel(R) Core(TM) i5-8250U
2. Memory : 8.00 GB
3. Storage : 128 GB SSD, 1 TB HDD
4. GPU : Geforce MX150

### 3.2.5. Deployment dan Migrasi *Smart Contract*

Dalam pengerjaan sistem ledger, *smart contract* akan di *deploy* pada jaringan *blockchain* Polygon Mumbaiuntuk dapat menjalankan transaksinya. Pada proses *deployment* memanfaatkan *tools* Brownieuntuk untuk menyiapkan kontrak pada jaringan *blockchain.* Proses awal untuk melakukan *deployment* kontrak adalah menjalankan *scripts* deploy.py dan memasukkan alamat API Alchemy pada file .env (*environment*). Setelah kontrak berhasil di *deploy* akan mendapatkan output berupa file map.json yang berisi *hash* alamat kontrak pada jaringan Polygon. Nantinya file tersebut juga yang akan digunakan *back-end* untuk menghubungkan ke jaringan Polygon. Adapun perintah dan *scripts* brownie yang digunakan adalah sebagai berikut :

**Tabel 3. 1** Perintah brownie

|  |  |
| --- | --- |
| **Perintah yang digunakan pada Brownie** | |
| Fungsi | Perintah |
| Menambahkan jaringan lokal ganache | brownie networks add Ethereum ganache-local host=http://127.0.0.1:7545 chainid=5777 |
| Menambahkan jaringan Polygon Mumbai | brownie networks add Polygon polygon-mumbai host=<URL API> chainid=80001 name="Mumbai Testnet (Alchemy)" |
| *Deploy* dan migrasi kontrak pada jaringan lokal | brownie run scripts/deploy.py --network=ganache-local |
| *Deploy* dan migrasi kontrak pada jaringan Polygon | brownie run scripts/deploy.py --network= polygon-mumbai |
| Untuk melakukan reset *deployment smart contract* | brownie run scripts/reset.py |

# 

# BAB IV PENGUJIAN DAN HASIL

## 4.1. Pengujian

Pada sistem ledger ini dilakukan beberapa proses pengujian untuk mengetahui apakah sistem dapat berjalan sesuai dengan yang diinginkan. Dari pengujian yang dilakukan dapat ditentukan tingkat keberhasilan dari sistem yang sudah dirancang. Pengujian ini dilakukan pada jaringan lokal dan jaringan *Polygon Mumbai* yang merupakan jaringan test dari *polygon*. Adapun beberapa parameter yang akan dilakukan pengujian adalah sebagai berikut :

1. Pengujian validitas fungsional menggunakan metode *white box*
2. Pengujian performansi *blockchain* dalam menjalankan transaksi
3. Pengujian validitas data setiap transaksi dengan menggunakan *polygon mumbai scan* untuk mengetahui data yang ditulis sudah sesuai*.*
4. Pengujian duplikasi data untuk mengetahui bagaimana jika dilakukan penyimpanan file dengan data yang sama.

## 4.2. Pengujian Validitas Fungsional

Pengujian validitas fungsional adalah pengujian kerja *smart contract* dalam menjalankan transaksi pada *blockchain* dan integrasi IPFS dengan sistem yang sudah dibuat. Dengan pengujian ini bertujuan untuk memastikan *code* dan fungsi *logic* pada sistem dapat berjalan sesuai harapan. Harapannya sistem yang berjalan memiliki minim *bug* dan *erorr*. Berikut adalah alur dari pengujian white box yang sudah dilakukan.

**Tabel 4. 1** Pengujian Deployment dan Migrasi Kontrak

|  |  |  |  |
| --- | --- | --- | --- |
| **Kasus dan Hasil Uji** | | | |
| **Data Input** | **Hasil yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Deployment Kontrak | Menambahkan network Ganache pada Brownie untuk proses kompilasi pada jaringan lokal | Berhasil menambahkan network ganache pada Brownie untuk proses kompilasi pada jaringan lokal | Diterima |
| *Deployment* kontrak pada jaringan lokal dapat dieksekusi tanpa adanya *error* | *Deployment* kontrak pada jaringan lokal berhasil dieksekusi tanpa adanya *error* | Diterima |
| Brownie dapat menampilkan kredensial hasil *deployment* pada jaringan lokal yang berupa *hash* kontrak | Brownie berhasil menampilkan hasil *deployment* pada jaringan lokal yang berupa *hash* kontrak | Diterima |
| Brownie dapat melakukan *testing* pada jaringan lokal tanpa adanya pesan *error* dan menampilkan rincian pemakaian *gas fee* | Brownie berhasil melakukan *testing* pada jaringan lokal tanpa adanya pesan *error* dan menampilkan rincian pemakaian *gas fee* | Diterima |
| Menambahkan API alchemy pada Brownie untuk proses kompilasi pada jaringan Polygon Mumbai | Berhasil menambahkan API alchemy pada Brownie untuk proses kompilasi pada jaringan Polygon Mumbai | Diterima |
| *Deployment* kontrak pada jaringan Polygon Mumbai dapat dieksekusi tanpa adanya *error* | *Deployment* kontrak pada jaringan Polygon Mumbai berhasil dieksekusi tanpa adanya *error* | Diterima |
| Brownie dapat menampilkan kredensial hasil *deployment* pada jaringan Polygon Mumbai yang berupa *hash* kontrak | Brownie berhasil menampilkan hasil *deployment* pada jaringan Polygon Mumbai yang berupa *hash* kontrak | Diterima |
| Brownie dapat melakukan *testing* pada jaringan Polygon Mumbai tanpa adanya pesan *error* dan menampilkan rincian pemakaian *gas fee* | Brownie berhasil melakukan *testing* pada jaringan Polygon Mumbai tanpa adanya pesan *error* dan menampilkan rincian pemakaian *gas fee* | Diterima |

Berdasarkan hasil pengujian *white box* pada *deployment* dan migrasi kontrak diatas, didapatkan hasil bahwa *smart contract* berhasil di *deploy* dan migrasi dengan baik pada jaringan lokal maupun jaringan uji polygon. Selain itu juga tidak ditemukan pesan *error* selama proses *deployment* sampai migrasi kontrak. Dengan ini maka proses *deploy* dan migrasi kontrak berjalan sesuai dengan yang diharapkan oleh penguji.

**Tabel 4. 2** Pengujian Fitur Kontrak Penyimpanan File

|  |  |  |  |
| --- | --- | --- | --- |
| **Kasus dan Hasil Uji** | | | |
| **Data Input** | **Hasil yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol Connectuntuk menghubungkan wallet | Dapat terhubung dengan wallet dan menampilkan informasi alamat serta jumlah token | Berhasil terhubung dengan wallet dan menampilkan informasi alamat serta jumlah token | Diterima |
| Memilih tombol Upload setelah menambahkan file | Tombol dapat ditekan dan memunculkan dialog konfirmasi transaksi pada metamask | Tombol berhasil ditekan dan memunculkan dialog konfirmasi transaksi pada metamask | Diterima |
| Melakukan konfirmasi transaksi penyimpanan pada metamask | Mendapatkan informasi gas fee dan melakukan konfirmasi dengan memilih tombol Accept | Berhasil mendapatkan informasi gas fee dan melakukan konfirmasi dengan memilih tombol Accept | Diterima |
| Mendapatkan informasi transaksi yang berupa *hash* file dan *hash* transaksi dengan *etherscan* | Memastikan bahwa transaksi berhasil tercatat pada *blockchain* dan seluruh informasi transaksi berhasil didapatkan dengan *etherscan* | Transaksi berhasil tercatat dan informasi transaksi berupa *hash* file dan *hash* transaksi berhasil didapatkan dengan *etherscan* | Diterma |
| Entry file berhasil ditampilkan | Mendapatkan informasi berupa nama file, tanggal *upload,* dan ukuran file. | Berhasil mendapatkan informasi berupa nama file, tanggal *upload,* dan ukuran file. | Diterima |

Berdasarkan pengujian fitur kontrak mengirim file diatas, penguji dapat menghubungkan akun metamask dan melakukan transaksi untuk penyimpanan file dengan melakukan konfirmasi pada dialog metamask. Hasil transaksi berhasil tercatat dengan baik pada blockchain dan langsung ditampilkan pada halaman website. Dari hasil ini disimpulkan bahwa transaksi penyimpanan file dapat berjalan dengan baik sebagaimana semtinya.

**Tabel 4. 3** Pengujian Fitur Kontrak Membaca dan Tracking File

|  |  |  |  |
| --- | --- | --- | --- |
| **Kasus dan Hasil Uji** | | | |
| **Data Input** | **Hasil yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol Connectuntuk menghubungkan wallet | Dapat terhubung dengan wallet dan menampilkan informasi alamat serta jumlah token | Berhasil terhubung dengan wallet dan menampilkan informasi alamat serta jumlah token | Diterima |
| Mendapatkan informasi file yang sudah disimpan | Menampilkan informasi file – file yang sudah disimpan sebelumnya | Berhasil menampilkan informasi file – file yang sudah disimpan | Diterima |
| Dapat membuka file yang sudah disimpan | Mendapatkan akses untuk membuka file yang berupa link CID file yang sudah tersimpan dengan cara melakukan klik pada nama file | Berhasil mendapatkan akses untuk membuka file yang berupa link CID file yang sudah tersimpan dengan cara melakukan klik pada nama file | Diterima |
| Dapat melakukan tracking file berdasarkan *hash* transaksi | Dapat melakukan tracking file berdasarkan *hash* transaksi yang didapatkan dan menampilkan seluruh rincian transaksi | Berhasil melakukan tracking file berdasarkan *hash* transaksi yang didapatkan dan menampilkan seluruh rincian transaksi | Diterima |

Berdasarkan hasil pengujian fitur kontrak membaca dan tracking file diatas, penguji dapat menampilkan file yang sudah disimpan dan juga dapat melakukan tracking file berdasarkan *hash* transaksi. File yang tersimpan pada IPFS juga dapat dibuka dengan akses link *content identifier (CID)* dan dapat ditampilkan dengan baik. Dengan ini dapat disimpulkan bahwa pembacaan dan *tracking* file dapat berjalan dengan baik sebagaimana mestinya.

## 4.3. Pengujian Performansi

Pada pengujian performansi digunakan adalah performansi sistem dalam melakukan penyimpanan file. Dalam melakukan pengujian performansi sistem akan diberikan beberapa skenario uji. Yang pertama adalah pengujian kecepatan input file sampai dengan konfirmasi metamask. Jadi cara kerja dari aplikasi ini adalah dengan mengambil *buffer file* terlebih dahulu selanjutnya akan dilanjutkan dengan konfirmasi penyimpanan file. Jadi aplikasi ini tidak serta merta menyimpan keseluruhan data dari file tersebut dikarenakan keterbatasan penyimpanan pada jaringan *peer to peer*. Pengujian ini dilakukan dengan melakukan penyimpanan 25 file dengan ekstensi dan ukuran yang berbeda. Hal ini bertujuan untuk mengukur seberapa cepat sistem memproses data yang masuk sampai dengan melakukan konfirmasi pada metamask. Adapun hasil dari pengujian pertama adalah sebagai berikut.

**Gambar 4. 1** Performa Konversi File dan Konfirmasi Metamask

Dari hasil pengujian pada Gambar 4.1 didapatkan bahwa waktu komputasi untuk melakukan konversi file cukup beragam. Disini konversi file untuk mengambil buffer ditentukan dari kompleksitas data file dan juga besaran file. Dari pengujian konversi file didapatkan waktu rata – rata sistem untuk melakukan komputasi adalah 4,69 detik dan waktu rata – rata file untuk dapat diverifikasi oleh metamask adalah 11,48 detik. Pada proses konversi waktu yang didapat sangat cepat karena komputasi terdapat pada *VM* yang digunakan untuk hosting web. Sehingga proses komputasi tidak berjalan secara lokal dan membuat proses menjasi lebih cepat.

## 4.4. Pengujian Validitas Data

Pengujian validitas data adalah pengujian untuk memastikan data yang terinput pada *blockchain* sudah sesuai. Skenario pengujiannya adalah disiapkan satu buah file yang akan diinput ke sistem. Setelah proses penyimpanan berhasil dilakukan maka akan muncul tabel nama file yang sebelunya telah disimpan. Dari tabel tersebut akan menghasilkan sebuah alamat yang akan digunakan untuk mengakses file yang tersimpan pada IPFS. Alamat tersebut merupakan *content indentifier (CID)* IPFS versi 0 yang menggunakan enkripsi *base58btc.* Adapun CID dari file yang berhasil disimpan adalah sebagai berikut.

**Tabel 4. 5** CID File Input

|  |  |
| --- | --- |
| File | CID |
| Rizal – CV.pdf | https://ipfs.infura.io/ipfs/QmWQ97f7K7FvYMx8xvmRoRoYgZ1Lv6iTfrbWnjh7TudPxA |

Dari tabel diatas terlihat bahwa CID dari file yang sudah kita input adalah QmWQ97f7K7FvYMx8xvmRoRoYgZ1Lv6iTfrbWnjh7TudPxA. Selanjutnya kita akan melihat hasil pencatatan pada *blockchain* meggunakan *TxHash* yang didapat dari riwayat transaksi pada metamask. Setelah mendapatkan TxHash dan melakukan pencarian transaksi menggunakan *polygon scan,* setelah mendapatkan rincian transaksi didapatkan hasil input data yang menggunakan format biner. Adapun hasil input datanya adalah sebagai berikut.

Graphical user interface

Description automatically generated with low confidence

Graphical user interface, text, application

Description automatically generated

Dari hasil pengujian diatas didapatkan hasil bahwa CID file pada IPFS sama dengan input data ditulis pada blockchain. Sehingga membuktikan bahwa validasi file sudah berjalan dengan baik. Dengan adanya CID dapat memudahkan user untuk melakukan validasi file dikarenakan apabila data dari file tersebut sudah dirubah/dimodifikasi maka CID file juga akan ikut berubah. Sehingga dapat digunakan untuk menghindari adanya manipulasi data pada file tersebut.

## 4.5. Pengujian Duplikasi Data

Pengujian ini digunakan untuk melihat apakah sistem dapat melakukan penyimpanan dua buah file yang sama. Skenario pengujian pertama dilakukan dengan menyimpan terlebih dahulu sebuah file dengan format txt. Lalu dilanjutkan dengan melakukan penyimpanan lagi file yang sebelumya sudah berhasil kita simpan. Hasilnya file tetap berhasil disimpan tetapi dengan CID file yang sama dengan file yang sebelumnya sudah disimpan.

Pengujian selanjutnya adalah dengan melakukan perubahan/modifikasi data dari file yang sebelumnya sudah tersimpan. Modifikasi ini dilakukan dengan merubah sedikit isi dari file tersebut dan dilakukan *save as* dengan nama file yang sama. Hal ini bertujuan untuk merubah metadata dari sebuah file biarpun memiliki nama yang sama. Selanjutnya adalah dilakukan penyimpanan lagi seperti sebelumnya, hasilnya adalah file berhasil disimpan pada IPFS. Tetapi file yang tersimpan memiliki CID yang berbeda dari dua file sebelumnya yang sudah kita simpan. Sehingga dapat disimpulkan bahwa pada IPFS tidak dapat melakukan penyimpanan file yang memiliki metadata yang sama atau duplikat. Tetapi apabila terdapat file dengan nama yang sama tetapi memiliki metadata yang berbeda maka akan berhasil disimpan.

# 

# BAB V SIMPULAN DAN SARAN

## 5.1. Simpulan

Bab ini memuat elaborasi dan rincian simpulan yang kemudian menjadi bagian abstraks. Simpulan ditarik dari hasil analisis secara komprehensif atas eksperimen yang telah dilakukan dan dinyatakan dalam bentuk narasi satu dua paragraf. Dalam Simpulan menggambarkan tingkat ketercapaian atas Tujuan Tugas Akhir yang telah dinyatakan dalam Bab 1.

## 5.2. Saran

Di dalam Saran, untuk pengembangan penelitian sebelumnya, pembuatan sistem disarankan untuk lebih xxx, sehingga tujuan dapat lebih tercapai. Alasannya :

1. Harus lebih mengidentifikasi masalah
2. Harus menyesuaikan dengan teknologi yang ada
3. Kelemahan-kelemahan yang terjadi

# DAFTAR PUSTAKA

1. B. N. Eason, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (*references*)
2. J. C. Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
3. I. S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

# LAMPIRAN

Lampiran dapat berisi kode sumber, tabel-tabel yang diperlukan dalam penelitian tapi kurang relevan untuk dimasukkan dalam bab-bab dalam Tugas Akhir.