

**APLIKASI INVENTORY ASET MENGGUNAKAN
JARINGAN *BLOCKCHAIN***



SKRIPSI

Diajukan sebagai salah satu syarat untuk menyelesaikan pendidikan diploma empat (D-4) program Studi Teknik Komputer Dan Jaringan
Jurusan Teknin Elektro
Politeknik Negeri Ujung Pandang

INRIATI
425 19 008

PROGRAM STUDI D-4 TEKNIK KOMPUTER DAN JARINGAN
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2023

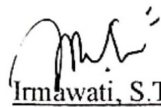
HALAMAN PENGESAHAN

Skripsi dengan judul **APLIKASI INVENTORY ASET MENGGUNAKAN JARINGAN BLOCKCHAIN** oleh **INRIATI** Nomor Induk Mahasiswa **42519008** telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar Diploma IV (D-4/S1 Terapan) pada Program Studi Teknik Komputer dan Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Makassar, 25 September 2023

Mengesahkan,

Pembimbing I



Irmawati, S.T.,M.T.
NIP. 19781124 201212 2 002

Pembimbing II



Ir. Dahlia, M.T.
NIP. 19641231 199103 2 003

Mengetahui,

Koordinator Program Studi
Teknik Komputer dan Jaringan



Eddy Tungadi, S.T.,M.T.
NIP. 19790823 201012 1 001

HALAMAN PENERIMAAN

Pada hari ini, Rabu tanggal 27 September 2023 Tim Penguji Sidang Skripsi telah menerima dengan baik hasil skripsi oleh mahasiswa: **INRIATI** Nomor Induk Mahasiswa 425 19 008 dengan judul “**APLIKASI INVENTORY ASET MENGGUNAKAN JARINGAN BLOCKCHAIN**”.

Makassar, 27 September 2023

Tim penguji Ujian Sidang Skripsi:

- | | | |
|--|---------------|---------|
| 1. Prof. Irfan Syamsuddin, S.T., M.Com.ISM., Ph.D. | Ketua | (.....) |
| 2. Meylanie Olivya, S.T., M.T. | Sekretaris | (.....) |
| 3. Tantri Indrabulan, S.T., M.T. | Anggota | (.....) |
| 4. Iin Karmila Yusri, S.ST., M.Eng., Ph.D | Anggota | (.....) |
| 5. Irmawati, S.T., M.T. | Pembimbing I | (.....) |
| 6. Ir.Dahlia, M.T. | Pembimbing II | (.....) |



KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulisan skripsi ini yang berjudul “APLIKASI INVENTARISASI ASET MENGGUNAKAN JARINGAN *BLOCKCHAIN* dapat diselesaikan dengan baik. Skripsi ini disusun guna memenuhi salah satu syarat untuk menyelesaikan studi serta dalam rangka memperoleh gelar Diploma IV (D-4/S1 Terapan) pada Program Studi Teknik Komputer dan Jaringan Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Penulis menyadari bahwa keberhasilan penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak baik secara langsung maupun tidak langsung. Oleh karena itu, dengan rendah hati penulis mengucapkan terima kasih kepada:

1. Kedua Orang tua penulis yakni Bapak Rudding dan Ibu Nuriati, Saudara penulis yakni Yurniati, Rikal, Rilgi, Alpin, dan Rifki yang selalu memberikan doa, semangat, dan motivasi kepada penulis.
2. Bapak Prof. Ir. Ilyas Mansyur, M.T. selaku Direktur Politeknik Negeri Ujung Pandang.
3. Bapak Ahmad Rizal Sultan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro.
4. Bapak Eddy Tungadi, S.T., M.T. selaku Kepala Program Studi Teknik Komputer dan Jaringan
5. Ibu Irmawati, S.T., M.T. sebagai Pembimbing I dan Ibu Ir. Dahlia, M.T. sebagai Pembimbing II yang telah mencurahkan waktu, ilmu, motivasi, nasehat dan segala arahan serta kesabarannya dalam membimbing penulis menyelesaikan skripsi ini.
6. Teman peneliti Muhammad Fadhil Abrar Lasawedi yang telah membantu dalam proses belajar peng-kodean.
7. Pegawai dan staff BPS terkhusus kepada Pak Elyas Bangapadang Musu, Ibu Bertin dan Kak Makdum yang telah bersedia bekerja sama dalam memberikan saran atau masukan terkait skripsi ini.
8. Teman-teman seperjuangan Angkatan 2019 Program Studi Teknik Komputer dan Jaringan khususnya kelas A yang telah berjuang bersama-sama selama kurang lebih 4 tahun.
9. Semua pihak yang telah memberikan dukungan moril maupun materil yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa skripsi ini jauh dari kesempurnaan, oleh sebab itu penulis ingin meminta maaf jika terdapat banyak kesalahan dalam penulisan skripsi ini. Akhir kata penulis mengharapkan adanya kritik dan saran yang sifatnya membangun untuk perbaikan dimasa mendatang. Semoga tulisan ini bermanfaat.

Makassar, 18 September 2023



Penulis

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
HALAMAN PENERIMAAN	iii
KATA PENGANTAR.....	iv
DAFTAR ISI	v
DAFTAR TABEL.....	viii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN	xi
SURAT PERNYATAAN	xii
RINGKASAN	xiii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Ruang Lingkup Penelitian.....	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
BAB 2 TINJAUAN PUSTAKA.....	5
2.1 Badan Pusat Statistik Provinsi Sulawesi Selatan	5
2.2 Sistem Inventarisasi Aset.....	5

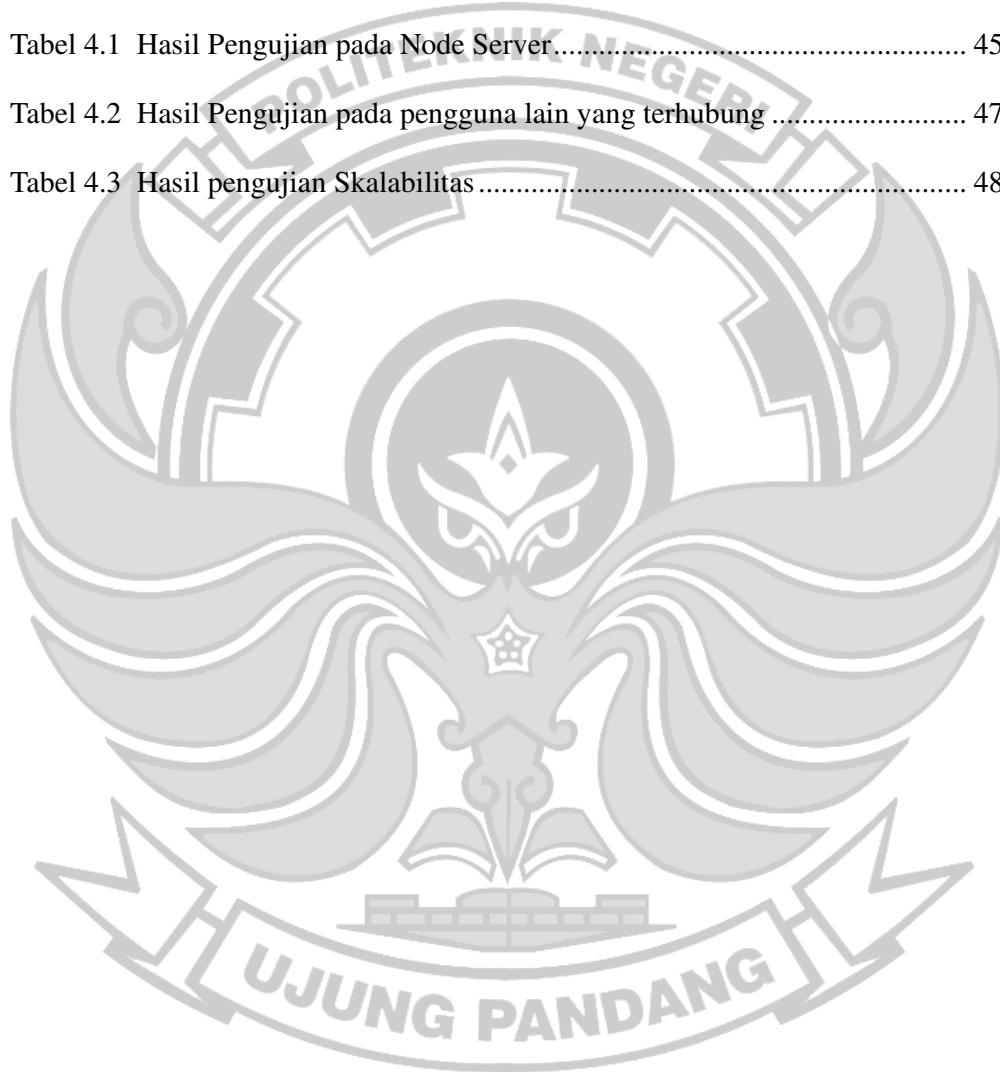
2.4 Blockchain.....	7
2.5 Smart Contract.....	10
2.6 MetaMask.....	11
2.7 Ganache.....	13
2.8 Solidity.....	14
2.9 Ethereum.....	15
2.10 Web3.....	15
2.11 Visual Studio Code.....	16
2.12 Pengujian Fungsionalitas dan Skalabilitas.....	17
BAB 3 METODE PENELITIAN.....	18
3.1 Tempat dan Waktu Penelitian.....	18
3.2 Alat dan Bahan.....	18
3.2.1 Alat.....	18
3.2.2 Bahan Penelitian.....	19
3.3 Prosedur Penelitian.....	20
3.3.1 Analisis Kebutuhan.....	20
3.3.2 Design Sistem.....	21
3.3.3 <i>Coding</i>	29
3.3.4 Implementasi dan Pengujian.....	30
BAB IV HASIL DAN PEMBAHASAN.....	33

4.1 Hasil Antarmuka WEB3	33
4.2 Pembuatan <i>Smart Contract</i> dengan <i>Blockchain</i> lokal Ganache	39
4.3 Menghubungkan MetaMask Pengguna Lain ke Ganache Server.....	43
4.4 Import Akun dari Ganache ke MetaMask.....	44
4.5 Pengujian Fungsionalitas.....	45
4.6 Pengujian Skalabilitas.....	48
BAB V PENUTUP.....	50
5.1 Kesimpulan.....	50
5.2 Saran.....	50
DAFTAR PUSTAKA.....	52
LAMPIRAN.....	54



DAFTAR TABEL

Tabel 2.1 Penelitian Terkait <i>Blockchain</i>	9
Tabel 3.1 Kebutuhan Perangkat Keras	18
Tabel 3.2 Kebutuhan Perangkat Lunak	18
Tabel 4.1 Hasil Pengujian pada Node Server.....	45
Tabel 4.2 Hasil Pengujian pada pengguna lain yang terhubung	47
Tabel 4.3 Hasil pengujian Skalabilitas	48



DAFTAR GAMBAR

Gambar 2.1	Cara Kerja <i>Blockchain</i>	8
Gambar 2. 2	Tampilan Awal MetaMask.....	12
Gambar 2. 3	Tampilan Awal Ganache.....	14
Gambar 3.1	Kategori Inventaris.....	19
Gambar 3.2	Tahapan Penelitian	20
Gambar 3.3	Gambaran Umum Database Tradisional	22
Gambar 3.4	Gambaran Umum Sistem <i>Blockchain</i>	22
Gambar 3.5	Flowchart yang diterapkan BPS.....	25
Gambar 3.6	Flowchart yang disarankan	26
Gambar 3.7	<i>Use Case Diagram</i>	27
Gambar 3. 8	<i>Sequence Diagram</i>	29
Gambar 4.1	Tampilan Login MetaMask.....	33
Gambar 4.2	Tampilan Register.....	34
Gambar 4.3	Tampilan Login	35
Gambar 4.4	Tampilan Halaman Index	35
Gambar 4.5	Tampilan Tambah Data Aset	36
Gambar 4.6	Halaman Tambah lebih dari satu data	36
Gambar 4.7	Halaman Edit Data	37
Gambar 4.8	Halaman Hapus Data.....	38
Gambar 4.9	Halaman Index untuk User.....	39
Gambar 4. 10	<i>Blockchain</i> Lokal Ganache.....	40
Gambar 4. 11	<i>Script Smart Contract</i> untuk inventarisasi	41

Gambar 4. 12 *Script Smart Contract* untuk login 42

Gambar 4. 13 Konfirmasi *New Network*..... 43

Gambar 4. 14 *Private Key* pada Akun Ganache..... 44

Gambar 4.15 Import Akun ke MetaMask 45

Gambar 4.16 Topologi..... 48



DAFTAR LAMPIRAN

Lampiran 1. Proses Instalasi Tools.....	54
---	----



SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Inriati

NIM : 42519008

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul “**APLIKASI INVENTORY ASET MENGGUNAKAN JARINGAN *BLOCKCHAIN***” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 18 September 2023



APLIKASI INVENTORY ASET MENGGUNAKAN JARINGAN *BLOCKCHAIN*

RINGKASAN

Kemajuan teknologi semakin berkembang dengan pesat. Begitu banyak perubahan dimana manusia mulai mengembangkan dan menggunakan teknologi yang begitu canggih dan praktis dalam berbagai hal. Misalnya dalam hal inventarisasi Barang Milik Negara (BMN). Barang Milik Negara (BMN) merupakan barang-barang yang diperoleh atau didapatkan dari negara menggunakan APBN atau berasal dari hal yang sah lainnya. Barang Milik Negara dapat dikategorikan sebagai aset. Dalam proses inventarisasi aset di Kantor BPS tentunya harus teliti dan dikelola dengan baik guna menghindari kesalahan dalam proses management aset. Masalah yang sering terjadi terkait sistem inventarisasi aset yaitu kesulitan mengetahui keterangan aset yang ada disana, seperti, sulitnya mengetahui riwayat, letak aset dan juga sulit mengetahui kondisi aset itu baik, rusak, maupun hilang. Pada penelitian ini, metode yang digunakan untuk inventarisasi aset yaitu menggunakan jaringan *Blockchain*. *Blockchain* merupakan sebuah sistem penyimpanan data digital yang terdiri dari banyak server (multiserver). Dimana Ganache sebagai Jaringan *Blockchain* local dan MetaMask digunakan untuk transaksi. Serta pembuatan *Smart Contract* untuk mengatur perilaku pengguna didalam aplikasi dan pembuatan aplikasi Web3 sebagai sarana demonstrasi penggunaan aplikasi. Sementara untuk prosedur penelitian dilakukan analisis kebutuhan, desain system, *coding*, serta implementasi dan pengujian. Hasil dari pengujian fungsionalitas aplikasi yaitu seluruh fitur didalam aplikasi sudah berjalan sesuai skenario yang diharapkan dan hasil untuk pengujian skalabilitas yaitu penggunaan jaringan *Blockchain* pada aplikasi ini mampu menangani jika terjadi lonjakan jumlah transaksi. Sehingga kesimpulan dari penelitian ini yaitu admin pengurus aset di Kantor BPS sudah bisa memmanagement aset dimana data aset yang di input tersimpan di jaringan *Blockchain* serta proses pencatatan asetnya tercatat dengan jelas.

Kata Kunci: Aset, BMN, BPS, Inventarisasi Aset, Jaringan *Blockchain*

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi semakin berkembang dengan pesat. Begitu banyak perubahan dimana manusia mulai mengembangkan dan menggunakan teknologi yang begitu canggih dan praktis dalam berbagai hal (Marpaung, 2018). Misalnya dalam hal inventarisasi Barang Milik Negara (BMN). Barang Milik Negara (BMN) merupakan barang-barang yang diperoleh atau didapatkan dari negara menggunakan APBN atau berasal dari hal yang sah lainnya. Barang Milik Negara dapat dikategorikan sebagai aset (Ratnawati, 2015). Salah satu instansi yang sangat memperhatikan inventarisasi aset yaitu kantor Badan Pusat Statistik Provinsi Sulawesi Selatan (BPS Prov. Sul Sel). Badan Pusat Statistik (BPS) adalah salah satu instansi yang difasilitasi oleh negara karena BPS merupakan lembaga Pemerintah Non-Kementerian yang bertanggung jawab langsung kepada Presiden (Maulana, 2020).

Dalam proses inventarisasi aset di Kantor BPS tentunya harus teliti dan dikelola dengan baik guna menghindari kesalahan dalam proses management aset. Masalah yang sering terjadi terkait sistem inventarisasi aset yaitu kesulitan mengetahui keterangan aset yang ada disana. Seperti, sulitnya mengetahui riwayat, letak aset dan juga sulit mengetahui kondisi asset itu baik, rusak, maupun hilang. Hal tersebut membuat proses inventarisasi menjadi tidak maksimal (Pambudi & Arvianto, 2016).

Masalah lainnya yang juga sering terjadi terkait inventarisasi aset yaitu pada proses pelacakan produk diseluruh rantai pasokan untuk memastikan bahwa kondisi

barang sampai ke pelanggan dalam keadaan baik, tidak dirusak dan dalam keadaan asli. Sehingga hal tersebut dapat berpengaruh pada keuntungan penjualan perusahaan (Mehta, 2021). Dan hal ini juga terjadi pada Kantor BPS untuk inventarisasi aset dimana diperlukan sebuah aplikasi yang memudahkan proses inventarisasi aset untuk memastikan keterangan aset.

Salah satu inovasi untuk meningkatkan pemeliharaan aset perusahaan yaitu dengan membuat aplikasi untuk inventarisasi aset menggunakan jaringan *Blockchain*. Dengan adanya aplikasi ini, perusahaan maupun instansi dapat dengan mudah menjamin mutu suatu informasi yang akan disajikan (Aminah et al., 2020).

Pada penelitian ini, metode yang digunakan untuk inventarisasi aset yaitu menggunakan jaringan *Blockchain*. *Blockchain* merupakan sebuah sistem penyimpanan data digital yang terdiri dari banyak server (multiserver). Para ahli juga mengungkapkan bahwa *Blockchain* adalah suatu sistem transaksi dan management data digital yang tersebar dimana semua pengguna sistem tersebut mempunyai satu consensus bersama (Fadhillah et al., 2022). Dengan membuat sistem secara tersebar, *Blockchain* menghilangkan perantara sehingga bisa membuat transaksi lebih murah.

Pemanfaatan *Blockchain* tak sebatas pada mata uang kripto. Sebab, teknologi yang berfungsi merekam data dan transaksi ini bisa digunakan untuk banyak sektor seperti pencatatan proses transaksi, dokumentasi, pemungutan suara, bahkan dalam proses inventarisasi aset (Nugraha, 2022).

Penelitian terkait dengan teknologi *Blockchain* sudah dilakukan oleh beberapa peneliti, salah satunya penelitian yang berjudul “*Blockchain*-Based Smart

Tracking and Tracing Platform for Drug Supply Chain”. Metode yang digunakan pada penelitian ini adalah *Blockchain* dengan consensus pBFT yang mampu meningkatkan efisiensi dan efektifitas pada proses regulasi. Penelitian ini menghasilkan sebuah sistem yang dapat melacak produksi obat-obatan, logistiknya, penjualannya, dan penggunaannya. Kemudian dapat menentukan apakah data obat-obatan tersebut dimasukkan kedalam rantai atau tidak, dan selanjutnya dapat menentukan solusi dari seleksi obat-obatan dengan kualitas rendah maupun palsu (Liu et al., 2021).

Kini *Blockchain* telah berkembang dengan pesat dan telah diimplementasikan dalam berbagai bidang. *Blockchain* awalnya diterapkan pada dunia bitcoin namun terjadi paradigma, dan konsep *Blockchain* dapat diimplementasi pada bidang akuntansi, audit, tracking asset, keuangan, *supply chain*, dan *public sector*. *Blockchain* dapat memberikan rekam jejak dari suatu barang dari tempat asal hingga ke tujuan yang dimana dapat membantu untuk mengetahui keterangan lengkap suatu barang. Oleh karena itu penulis memutuskan untuk mencoba mengimplementasikan *Blockchain* dalam inventarisasi aset di Kantor BPS.

1.2 Rumusan Masalah

Berdasarkan latar belakang penelitian di atas maka masalah yang dapat dirumuskan adalah bagaimana membangun Aplikasi Inventarisasi Aset menggunakan jaringan *Blockchain*?

1.3 Ruang Lingkup Penelitian

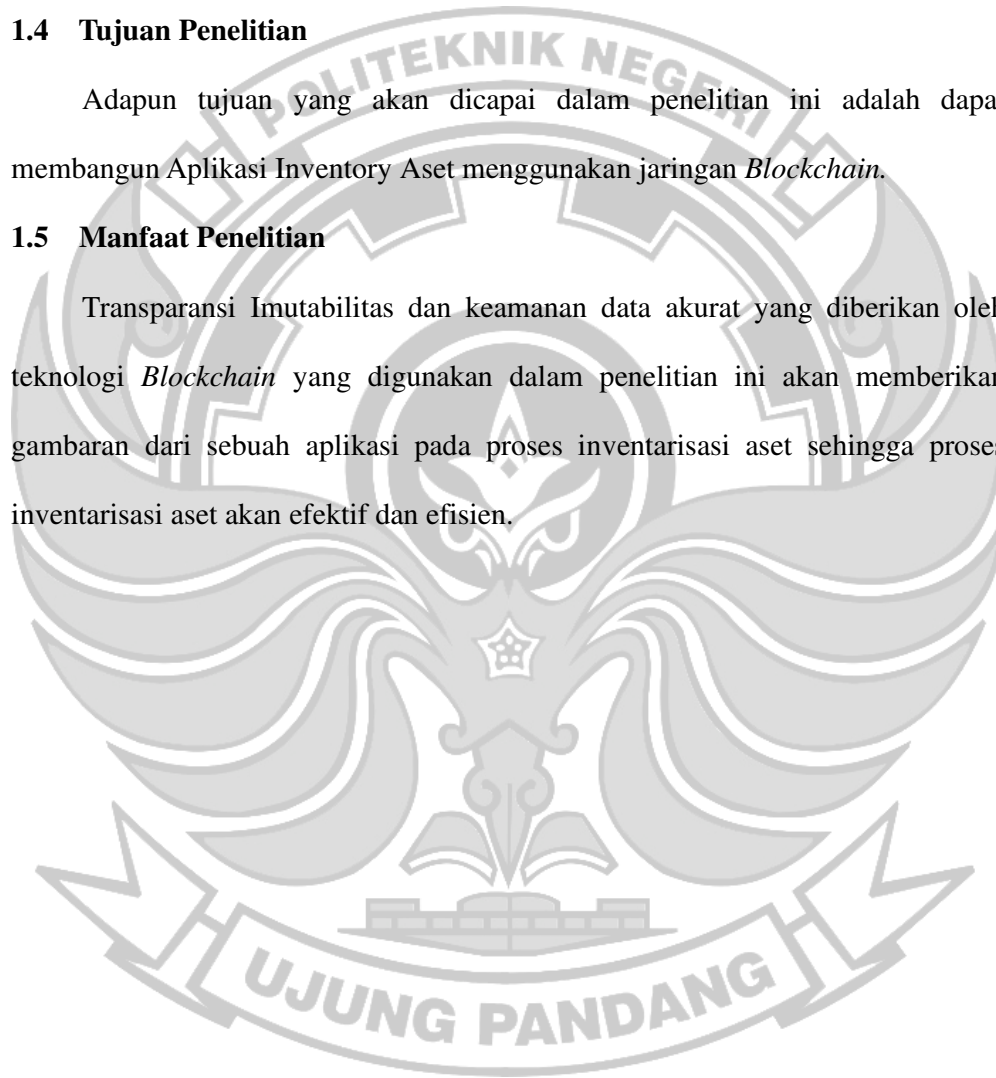
Penelitian ini akan berfokus pada sistem Inventarisasi Aset dengan menerapkan teknologi jaringan *Blockchain* menggunakan simulasi *software* Ganache.

1.4 Tujuan Penelitian

Adapun tujuan yang akan dicapai dalam penelitian ini adalah dapat membangun Aplikasi Inventory Aset menggunakan jaringan *Blockchain*.

1.5 Manfaat Penelitian

Transparansi Imutabilitas dan keamanan data akurat yang diberikan oleh teknologi *Blockchain* yang digunakan dalam penelitian ini akan memberikan gambaran dari sebuah aplikasi pada proses inventarisasi aset sehingga proses inventarisasi aset akan efektif dan efisien.



BAB 2 TINJAUAN PUSTAKA

2.1 Badan Pusat Statistik Provinsi Sulawesi Selatan

Badan Pusat Statistik adalah Lembaga Pemerintah Non Kementrian yang bertanggung jawab langsung kepada Presiden. Sebelumnya, BPS merupakan Biro Pusat Statistik, yang dibentuk berdasarkan UU Nomor 6 Tahun 1960 tentang Sensus dan UU Nomor 7 Tahun 1960 tentang Statistik. Sebagai pengganti kedua UU tersebut ditetapkan UU Nomor 16 Tahun 1997 tentang Statistik. Berdasarkan UU ini yang ditindaklanjuti dengan peraturan perundangan dibawahnya, secara formal nama Biro Pusat Statistik diganti menjadi Badan Pusat Statistik (BPS, 2021).

Berdasarkan Undang – undang, peranan yang harus dijalankan oleh BPS adalah sebagai berikut:

- a. Menyediakan kebutuhan data bagi pemerintah dan masyarakat.
- b. Membantu kegiatan statistik di kementrian, lembaga pemerintah atau instansi lainnya.
- c. Mengembangkan dan mempromosikan standar teknik dan metodologi statistik.
- d. Membangun kerja sama dengan institusi internasional dan negara lain untuk kepentingan perkembangan statistik Indonesia.

2.2 Sistem Inventarisasi Aset

Inventaris adalah suatu daftar yang di dalamnya berisi sumber daya penting dan aset yang dimiliki perusahaan yang dimanfaatkan untuk keberlangsungan perusahaan secara umum. Inventarisasi Aset adalah semua kegiatan dan usaha untuk memperoleh data yang diperlukan tentang ketersediaan barang-barang yang dimiliki dan diurus, baik yang diadakan melalui pembelian menggunakan anggaran

belanja, maupun sumbangan atau hibah untuk diadministrasikan sebagaimana mestinya menurut ketentuan dan cara yang telah ditetapkan di masing-masing instansi (Novendri et al., 2019). Undang - Undang Nomor 1 Tahun 2004 juga menjelaskan mengenai aset atau barang milik negara yaitu semua barang yang dibeli atau diperoleh atas beban APBN dan perolehan lainnya yang sah (Nasrudin, 2015). Aplikasi Inventarisasi Aset digunakan untuk manajemen aset perusahaan. Sistem ini dapat membantu perusahaan dalam mengetahui riwayat aset dan memudahkan manajemen aset (Saputro & Chandra, 2021).

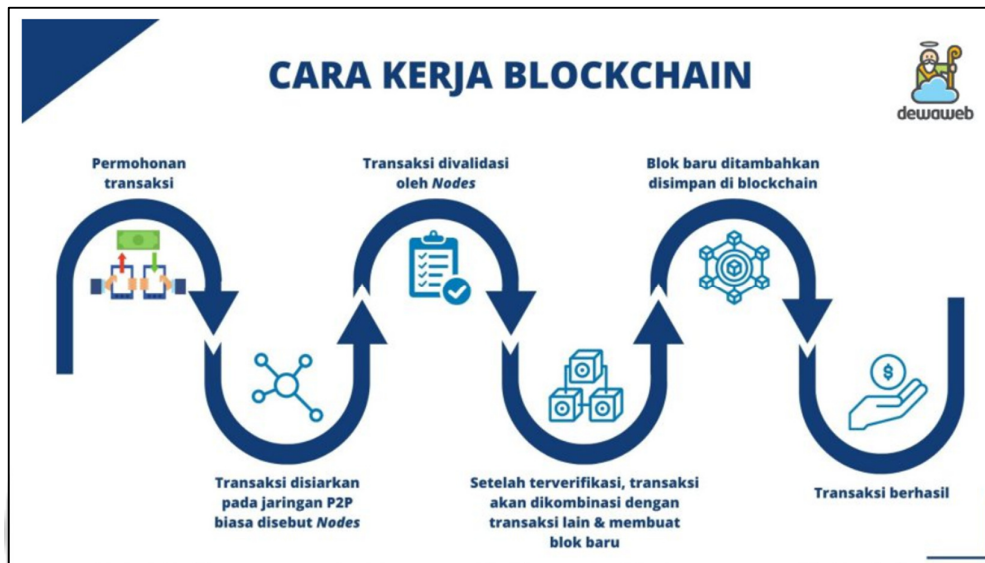
Tentu Badan Pusat Statistik provinsi Sulawesi Selatan memiliki aset-aset sebagai penunjang kegiatan instansi seperti meja, kursi, papan tulis, komputer, printer, dan yang lainnya. Saat ini BPS belum memiliki sistem internal yang dapat mengontrol aset-aset yang dimiliki tersebut tentu menyebabkan kesulitan dalam pengelolaan data-data aset dalam merencanakan, memperbaiki maupun pada saat membeli aset baru. Permasalahan tersebut dapat menyebabkan pengelola aset kesulitan dalam mencatat keterangan aset seperti melakukan pendataan riwayat berupa lokasi dan pendataan kondisi aset itu baik, rusak, ataupun dalam keadaan hilang dan pengelompokan aset berdasarkan jenisnya. Hal tersebut dapat menyebabkan proses laporan aset serta penggantian aset yang rusak pada BPS ini menjadi tidak maksimal. Solusi untuk mengatasi permasalahan yang ada, tentunya perlu sebuah aplikasi dalam pengelolaan aset yang dapat memudahkan manajemen dalam mengelola aset menjadi lebih efisien, efektif, dan terstruktur.

2.4 Blockchain

Blockchain atau dapat disebut juga sebagai teknologi pembukuan terdistribusi (Distributed Ledger Technology / DLT) merupakan sebuah konsep dimana setiap peserta atau pihak yang tergabung dalam jaringan terdistribusi memiliki hak akses terhadap pembukuan tersebut. Konsep yang dibawa oleh *Blockchain* merupakan penerapan konsep yang sudah ada, yaitu konsep database terdistribusi. Konsep ini lahir bersamaan dengan lahirnya bitcoin sekaligus sebagai jawaban atas permasalahan tidak adanya pihak ketiga (institusi finansial atau pemerintah) untuk membangun kepercayaan diantara pihak-pihak yang melakukan transaksi dilindungi yang tidak aman. Secara konseptual, teknologi *Blockchain* dapat disamakan dengan teknologi yang digunakan pada basis data terdistribusi. Dalam basis data terdistribusi informasi yang tercatat akan disimpan dan dibagikan kepada setiap anggota di jaringan tersebut (Noorsanti et al., 2018).

Blockchain disalin dan disebar di jaringan komputer. Setiap terjadinya transaksi baru atau blok baru maka akan ditambahkan ke dalam *Blockchain* dan setiap komputer yang ada pada jaringan tersebut akan memperbarui blokchainnya untuk mengikut perubahan. Data yang disimpan dalam *Blockchain* akan menyebar di antara beberapa node jaringan sehingga dapat menjaga keaslian data yang tersimpan di dalamnya. Jika ada salah satu pihak yang berusaha mengubahnya pada salah satu database, node lain tidak akan ikut berubah sehingga akan mencegah pihak tersebut melakukan tindak kejahatan. Jika ada salah satu transaksi yang rusak, maka setiap node akan saling memeriksa sehingga dapat dengan mudah

menunjukkan node dengan informasi yang salah. Ilustrasi terkait cara kerja *Blockchain* dapat dilihat pada gambar 2.1



Gambar 2.1 Cara Kerja *Blockchain*

Sumber: <https://www.dewaweb.com/blog/apa-itu-Blockchain/>

Perbedaan *Blockchain* dengan basis data yaitu pada cara bagaimana data terstruktur. *Blockchain* mengumpulkan informasi bersama dalam kelompok dan dikenal sebagai blok yang menyimpan berbagai kumpulan informasi. Blok memiliki kapasitas penyimpanan tertentu dan ketika diisi ke rantai blok yang sebelumnya telah terisi dan akan membentuk rantai data yang dikenal sebagai istilah *Blockchain*. Sehingga semua *Blockchain* adalah database namun tidak semua database adalah *Blockchain* (Yusup et al., 2019).

Blockchain memberikan transparansi imutabilitas dan keamana data yang akurat bagi organisasi atau instansi yang ingin membuat aplikasi Inventarisasi Aset dan memberikan solusi logistik. Teknologi ini praktis untuk rantai pasokan. Semua pihak terkait dalam rantai pasokan atau aliran logistik dapat mengikuti transaksi

dari awal hingga akhir. Kemampuan ini sangat penting bagi perusahaan dan pelanggan mereka. Teknologi *Blockchain* adalah buku besar terdistribusi, artinya mereka yang terlibat dalam jaringan dapat melihat informasi yang relevan mengenai perjanjian bisnis. Catatan setiap transaksi tetap berada dalam satu database. Setiap anggota jaringan memiliki akses ke database ini. Setelah aset dibeli dan ditransfer dari satu pihak ke pihak lain, transaksi tersebut dapat dilihat oleh semua pihak. Transparansi ini memungkinkan komunikasi yang jelas dan transaksi berkualitas lebih tinggi.

Kini *Blockchain* telah berkembang dengan pesat dan telah diimplementasikan dalam berbagai bidang. konsep *Blockchain* dapat diimplementasi pada bidang akuntansi, audit, Inventarisasi Aset, keuangan, *supply chain*, dan *public sector*. Penelitian terkait dengan teknologi *Blockchain* sudah dilakukan oleh beberapa peneliti sebelumnya. Penelitian terkait *Blockchain* dapat dilihat pada tabel 2.1

Tabel 2.1 Penelitian Terkait *Blockchain*

No	Judul	Metode	Masalah	Hasil
1	Aplikasi pendonasian barang berbasis web dengan teknologi <i>Blockchain</i> modul volunteer (Desember, 2020)	System Developme nt Life Cycle	Bantuan donasi sering tidak tepat sasaran, bantuan tersebut tidak sampai pada penerima bantuan serta yang membutuhkan dan jenis bantuan tidak sesuai dengan yang dibutuhkan	Aplikasi pendonasian barang berbasis web
No	Judul	Metode	Masalah	Hasil
2	Penerapan Teknologi	Menggunakan	Pelacakan dan ketertelusuran produk,	Sistem informasi

	<i>Blockchain</i> pada sistem <i>Supply Management</i> yang terintegrasi dengan sensor RFID (April, 2022)	pendekatan Study Literatur	transparansi keadaan atau kondisi produk dan keamanan data pada supply chain management yang dijalankan	yang terbaca melalui sensor RFID
3	Implementasi <i>Blockchain</i> dalam aplikasi pemilu (Agustus, 2021)	Metode <i>Waterfall (Linear Sequential Model)</i>	Manipulasi sering terjadi pada akhir pemilu.	Aplikasi pemilu berbasis web
4	Pemanfaatan Teknologi <i>Blockchain</i> pada program sertifikasi dosen (Jurnal Seminar Nasional, 2019)	<i>Script</i> Validasi <i>Blockchain</i>	Sering terjadi pemalsuan serta manipulasi sertifikat	E- <i>Certificate</i> yang diterapkan di ICM

2.5 Smart Contract

Smart Contract merupakan *Peace of Code* atau kode perjanjian antara dua orang dalam jaringan *Blockchain* yang berisi syarat dan ketentuan khusus. *Smart Contract* dijalankan secara otomatis tanpa memerlukan perantara. Transaksi hanya akan terjadi ketika syarat dan ketentuan khusus (*specific terms and conditions*) ini bertemu di antara dua node, tanpa melibatkan adanya intervensi dari pihak ketiga. Hal ini bisa terjadi karena seluruh proses dari *Smart Contract* ini dilakukan secara otomatis dalam sistem *Blockchain*. Untuk membuat *Smart Contract*, kita membutuhkan kode khusus yang sesuai dengan token kripto yang akan kita buat.

Kode ini harus dimasukkan ke dalam jaringan *Blockchain*, seperti *Ethereum* agar dapat bekerja dengan baik. Setelah kode siap, kita dapat melakukan *deploy contract* melalui *Ethereum wallet*.

Smart contract yang diterapkan pada *Blockchain* akan membuat transaksi dapat dilacak, transparan, dan tidak dapat diubah. Selain itu, *Smart Contract* semakin membuat teknologi *Blockchain* lebih kuat dan menjadikannya lebih cepat, lebih aman, terukur, dan terjangkau. Dalam rangkaian *Blockchain*, smart contract bekerja dengan cara mengikuti serangkaian aturan dan logika yang telah ditentukan di dalamnya. Setelah *Smart Contract* dibuat dan diunggah ke dalam jaringan *Blockchain*, kontrak tersebut akan otomatis dieksekusi saat semua syarat dan ketentuan terpenuhi. *Smart Contract* bisa dipandang seperti “kesepakatan pandai” yang dapat mengeksekusi transaksi secara otomatis tanpa perlu melibatkan pihak ketiga (Dewanto, 2021).

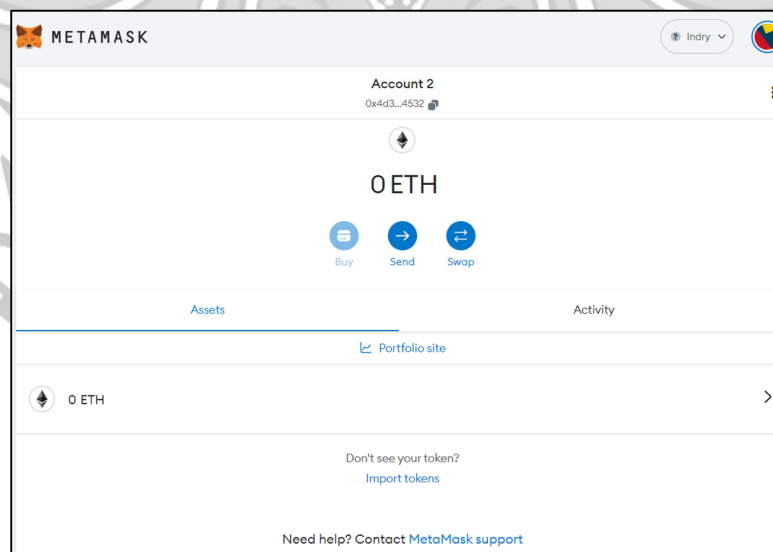
Bahasa pemrograman yang paling umum digunakan dalam pembuatan *Smart Contract* yaitu bahasa pemrograman *Sollidity*. memiliki sintaks yang mirip dengan JavaScript dan C++, sehingga memungkinkan pengembang untuk membuat perjanjian pintar yang kompleks dan efisien. Solidity mendukung konsep seperti variabel, tipe data, fungsi, dan struktur pengendalian yang umum ditemukan dalam pemrograman pada umumnya.

2.6 MetaMask

MetaMask adalah dompet (wallet) yang berbasis extension yang berjalan pada browser (Chrome, Firefox, Opera, atau Brave Browser) dapat dengan mudah digunakan dan nyaman untuk pengujian karena dapat terhubung ke berbagai node

Ethereum untuk menguji *Blockchain*. MetaMask merupakan dompet (wallet) Ethereum berbasis web. MetaMask bisa juga disebut portal bagi web2 dan Web3 sehingga biasa disebut "*Web3 provider*". Dengan kata lain, MetaMask memungkinkan penggunanya untuk menyimpan data terkait Ethereum seperti alamat *Public* dan *Private key* seperti dompet Ethereum lainnya, dan memungkinkan pengguna untuk berinteraksi dengan situs web yang menjalankan aplikasi berbasis Ethereum dan *Smart Contract* (web browser menjadi Ethereum browser) (Shell, 2016).

Untuk terhubung ke jaringan *Blockchain*, MetaMask perlu diarahkan ke titik akhir RPC. Pengguna dapat menambahkan jaringan *Blockchain* yang berbeda ke MetaMask dengan membuka "Pengaturan" dan memilih "Jaringan", lalu menekan tombol "Tambahkan Jaringan". Berikut dapat dilihat pada gambar 2.2 tampilan awal dari MetaMask

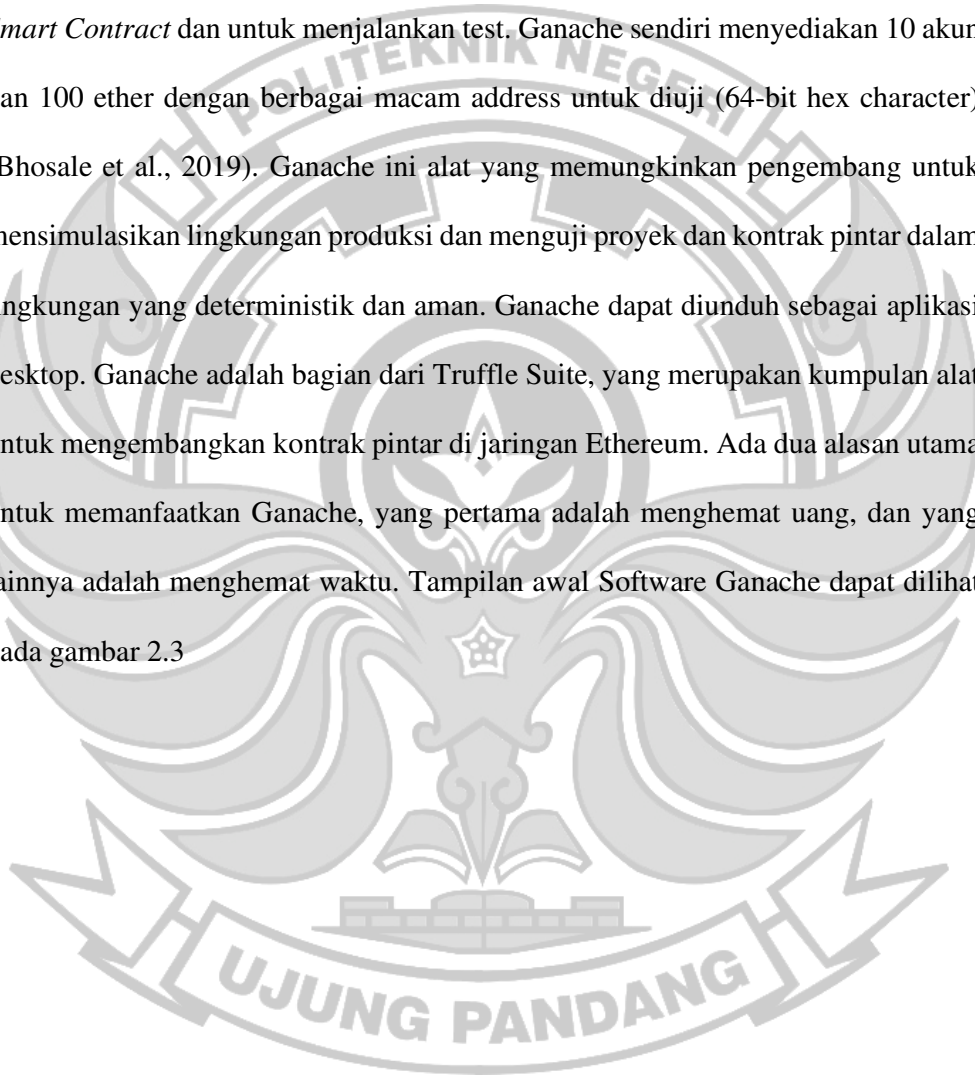


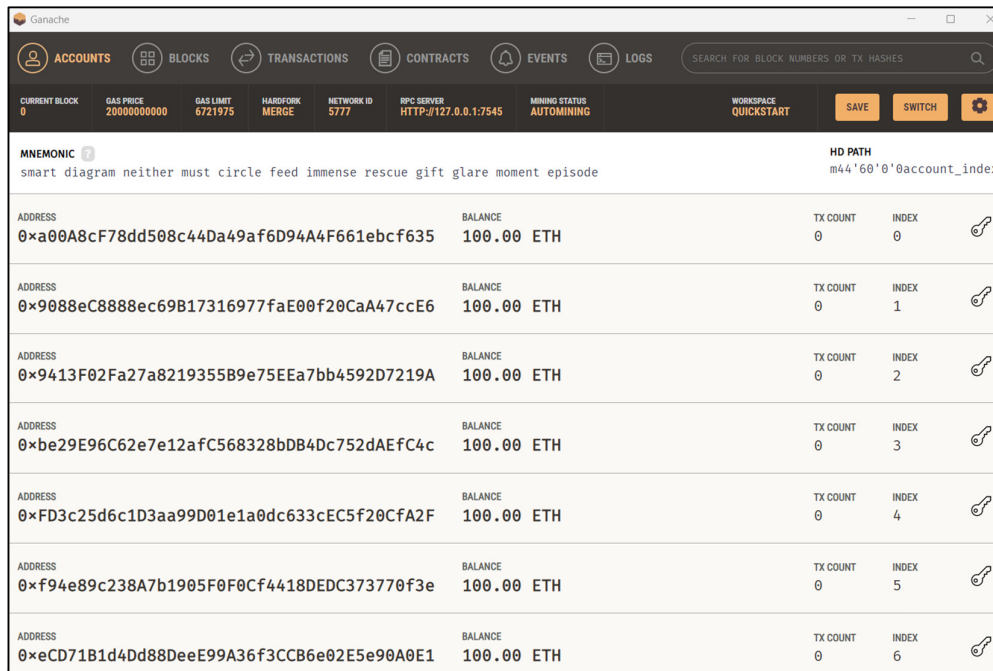
Gambar 2.2 Tampilan Awal MetaMask

Sumber: (Screenshoot tampilan awal MetaMask)

2.7 Ganache

Ganache adalah software yang digunakan untuk pengembangan Ethereum atau perangkat lunak simulasi yang memungkinkan pengembang *Blockchain* untuk membuat jaringan *Blockchain* lokal. Ganache juga digunakan untuk menyebarkan *Smart Contract* dan untuk menjalankan test. Ganache sendiri menyediakan 10 akun dan 100 ether dengan berbagai macam address untuk diuji (64-bit hex character) (Bhosale et al., 2019). Ganache ini alat yang memungkinkan pengembang untuk mensimulasikan lingkungan produksi dan menguji proyek dan kontrak pintar dalam lingkungan yang deterministik dan aman. Ganache dapat diunduh sebagai aplikasi desktop. Ganache adalah bagian dari Truffle Suite, yang merupakan kumpulan alat untuk mengembangkan kontrak pintar di jaringan Ethereum. Ada dua alasan utama untuk memanfaatkan Ganache, yang pertama adalah menghemat uang, dan yang lainnya adalah menghemat waktu. Tampilan awal Software Ganache dapat dilihat pada gambar 2.3





Gambar 2.3 Tampilan Awal Ganache
 Sumber: (Screenshoot aplikasi Ganache)

2.8 Solidity

Solidity adalah bahasa berorientasi objek (*Object Oriented Programming*) yang dirancang untuk menulis *Smart Contract* dalam Ethereum dan merupakan bahasa tingkat tinggi serta bisa dikatakan sebagai bahasa berorientasi kontrak (*Contract Oriented Programming*) dengan eksperimental pertama. *Solidity* ditulis secara statis, mendukung *inheritance* (pewarisan) library, dan tipe kompleks yang ditentukan pengguna di antara fitur-fitur lainnya. Bahasa pemrograman *Solidity* mirip dengan *Javascript* karena pada *Javascript* terdapat *class*, *class* yang terdapat pada *soldity* disebut *contract*. Syntax dan function nya pun banyak kemiripan dengan *Javascript* (Shell, 2016).

Dalam pengembangan *Blockchain*, *Solidity* menjadi bahasa pemrograman yang sangat penting karena digunakan untuk membuat *Smart Contract* yang

memungkinkan pengguna untuk membuat transaksi tanpa perlu melibatkan pihak ketiga. *Smart Contract* ini memungkinkan pengguna untuk membuat aplikasi terdesentralisasi yang aman dan transparan. Oleh karena itu, *Solidity* menjadi bahasa pemrograman yang sangat penting bagi para pengembang *Blockchain* dan aplikasi terdesentralisasi (Sutopo, 2023).

2.9 Ethereum

Ethereum adalah sebuah platform perangkat lunak terdesentralisasi yang memungkinkan *Smart Contracts and Distributed Applications* (DApps) dibangun dan dijalankan tanpa waktu henti, penipuan, kontrol, atau gangguan dari pihak ketiga. Ethereum juga merupakan teknologi untuk membangun aplikasi dan organisasi, menyimpan aset, melakukan transaksi, dan berkomunikasi tanpa dikendalikan oleh otoritas pusat. Ether (ETH) adalah mata uang kripto yang digunakan di platform Ethereum. Ethereum diciptakan oleh Vitalik Buterin, seorang remaja berusia 19 tahun yang tertarik pada sistem *Blockchain*.

Ethereum juga merupakan salah satu bentuk pengembangan dari teknologi *Blockchain* yang awalnya diterapkan pada Bitcoin. Ethereum dikembangkan untuk memungkinkan pengerjaan komputasi yang lebih kompleks pada framework *Blockchain* daripada hanya sekedar komputasi pencatatan data transaksi. Ethereum telah dikenal luas sebagai framework dalam mengembangkan *decentralized application* (Alexander Sugiharto et al., 2022).

2.10 Web3

Web3 atau Web 3.0 adalah perkembangan terbaru dari *World Wide Web* (WWW) yang menggabungkan konsep-konsep seperti desentralisasi, teknologi

Blockchain, dan ekonomi berbasis token. Web3 memungkinkan pengguna untuk berinteraksi dan berbisnis dengan pengguna lain secara langsung tanpa perantara pihak ketiga seperti bank atau social media. Web3 juga akan menggunakan *Machine Learning* dan *Artificial Intelligence* (AI) untuk memberdayakan platform yang lebih cerdas sekaligus adaptif. Web3 adalah generasi ketiga dari evolusi teknologi web yang berbasis *Blockchain* dan memiliki sistem yang terdesentralisasi.

Web3 dalam konteks *Blockchain* menggambarkan evolusi internet menuju model yang lebih desentralisasi dan pengguna-berpusat, di mana teknologi *Blockchain* dan kontrak pintar memainkan peran sentral. Web3 memungkinkan pengguna memiliki kontrol lebih besar atas data dan aset digital mereka, mendorong transaksi peer-to-peer tanpa perantara, dan memastikan keamanan serta privasi melalui kriptografi. Dengan fokus pada identitas digital, tokenisasi, dan interoperabilitas, konsep Web3 merintis era baru dalam pengembangan aplikasi, menciptakan potensi untuk perubahan fundamental dalam berbagai aspek interaksi online dan inovasi bisnis (Fidiana et al., 2018).

2.11 Visual Studio Code

Visual Studio Code (VSCoDe) adalah perangkat lunak penyunting kode sumber gratis yang dikembangkan oleh *Microsoft* untuk *Linux*, *macOS*, dan *Windows*. VSCoDe menyediakan berbagai fitur seperti penyorotan sintaksis, penyelesaian kode, kutipan kode, merefaktor kode, pengawakutuan, dan Git. VSCoDe juga mendukung hampir semua sistem operasi seperti *Windows*, *Mac OS*, *Linux*, dan lainnya. VSCoDe memiliki keunggulan dalam kelengkapan fitur dan

ekstensi yang membuatnya menjadi pilihan utama para pengembang. Selain itu, *VSCode* juga memiliki tampilan yang menarik dan mudah digunakan. *VSCode* juga mendukung banyak bahasa pemrograman seperti *C++*, *Python*, *Java*, *JavaScript*, *Solidity* dan bahasa pemrograman lainnya.

2.12 Pengujian Fungsionalitas dan Skalabilitas

Fungsionalitas adalah kemampuan dalam perangkat lunak digunakan untuk penyediaan fungsi-fungsi yang memenuhi kebutuhan user (Supriyono, 2019). Pengujian fungsionalitas aplikasi adalah proses yang berulang yang dapat memastikan bahwa aplikasi dapat berfungsi dengan baik, memenuhi kebutuhan pengguna dan bebas dari *Bug* yang dapat mengganggu pengalaman pengguna aplikasi. Tujuan utama dalam pengujian fungsionalitas yaitu untuk memastikan bahwa aplikasi tersebut berfungsi sebagaimana mestinya sesuai dengan spesifikasi yang telah ditentukan.

Skalabilitas adalah kemampuan jaringan untuk menangani jumlah transaksi yang semakin besar. Pengujian skalabilitas pada jaringan *Blockchain* dapat dilakukan dengan berbagai cara, seperti menghubungkan *Blockchain* yang berbeda untuk memanfaatkan kekuatan beberapa jaringan atau membangun platform *Blockchain* yang mendukung beberapa transaksi per detik (Apriliasari & Seno, 2022).

BAB 3 METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini akan dilakukan di Kantor Badan Pusat Statistik Provinsi Sulawesi Selatan dan di Lab Sistem Informasi Kampus 1 Politeknik Negeri Ujung Pandang pada bulan Maret 2023 sampai Agustus 2023.

3.2 Alat dan Bahan

3.2.1 Alat

1. Kebutuhan Perangkat Keras (*Hardware*)

Tabel 3.1 Kebutuhan Perangkat Keras

No	Nama Perangkat	Spesifikasi	Keterangan
1	Laptop/PC	<i>Processor Quad Core, Memory 8GB, HDD 500.</i>	Keperluan untuk pengkodean, simulasi, dan menjalankan perangkat lunak (<i>software</i>) yang dibutuhkan

2. Kebutuhan Perangkat Lunak (*Software*)

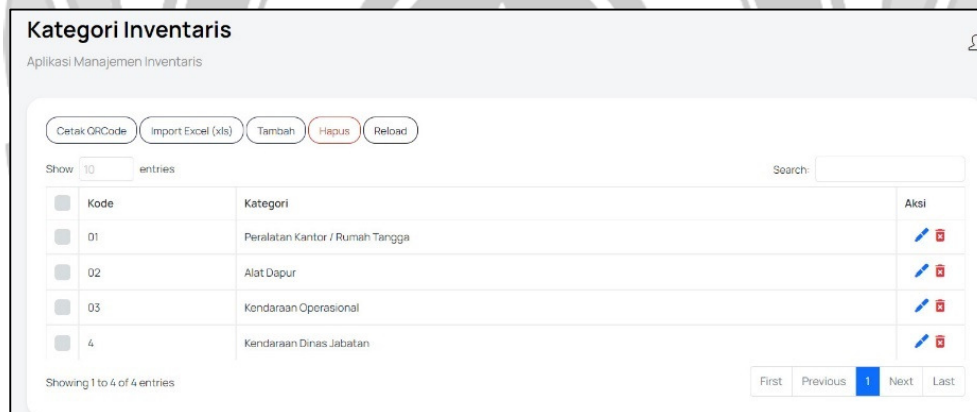
Tabel 3.2 Kebutuhan Perangkat Lunak

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Windows</i>	<i>Windows 10</i>	Sistem Operasi untuk menjalankan aplikasi
2	<i>WEB Browser</i>	<i>Google Chrome</i>	Untuk menjalankan aplikasi extension MetaMask
3	<i>Ganache Truffle</i>	<i>V2.7.0</i>	Untuk keperluan simulasi
4	MetaMask	Versi rilis 8.0.32	Untuk menghubungkan WEB3 dengan <i>Blockchain</i>
5	<i>Solidity</i>	Versi 0.8	Bahasa Pemrograman yang digunakan

No	Nama Perangkat	Spesifikasi	Keterangan
6	<i>Microsoft Visual Studio Code</i>	Versi 1.76.1	Untuk melakukan Pengodingan
7	<i>WEB3</i>	Versi 3.0	Untuk menyajikan tampilan website yang sudah disimulasikan oleh <i>Blockchain</i>

3.2.2 Bahan Penelitian

Bahan penelitian yang digunakan dalam penelitian ini berupa data dari aplikasi sistem inventarisasi aset di kantor BPS dimana data tersebut berupa file excel yang terdiri dari kolom nama, kategori, lokasi, dan kondisi. Pembuatan aplikasi ini mengacu pada aplikasi sebelumnya seperti pada gambar berikut:

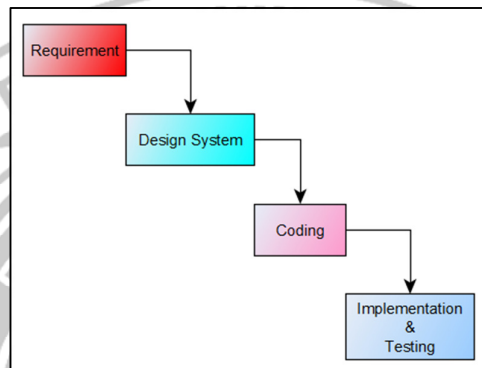


Gambar 3.1 Kategori Inventaris

Dilihat pada gambar 3.1 Kategori aset pada aplikasi inventarisasi aset di Kantor BPS dimana kategori tersebut terdiri dari 4 macam kategori yaitu Peralatan Kantor seperti Printer, PC dll. Alat Dapur seperti Lemari Es, Meja dll. Kendaraan Operasional seperti Yamaha Gear, Honda Vario dll. Dan Kendaraan Dinas Jabatan seperti Toyota Inova, Fino Sporti dll.

3.3 Prosedur Penelitian

Agar penelitian terstruktur dan berjalan dengan baik, maka diperlukan sebuah prosedur penelitian sehingga hasil yang diharapkan sesuai dengan tujuan penelitian. Penjelasan mengenai tahapan proses penelitian dapat dilihat pada gambar 3.2



Gambar 3.2 Tahapan Penelitian

3.3.1 Analisis Kebutuhan

Dalam tahap ini melakukan identifikasi kebutuhan sistem yang sesuai dengan harapan pengguna dan menganalisa sistem yang sedang berjalan saat ini untuk menganalisa kebutuhan guna mendapatkan solusi dari permasalahan. Data-data yang dibutuhkan pada tahapan ini biasanya diperoleh melalui wawancara maupun observasi.

Pada tahap ini dilakukan identifikasi masalah untuk mengetahui masalah yang terjadi pada proses inventarisasi aset di Kantor BPS, melalui wawancara langsung dengan ibu Bertin (Pegawai Bagian Umum) dan pak Elyas (yang menyarankan pembuatan sistem), kemudian melakukan studi literatur dengan melihat dan mencari hal – hal yang dibutuhkan oleh sistem.

Berikut masalah yang didapatkan:

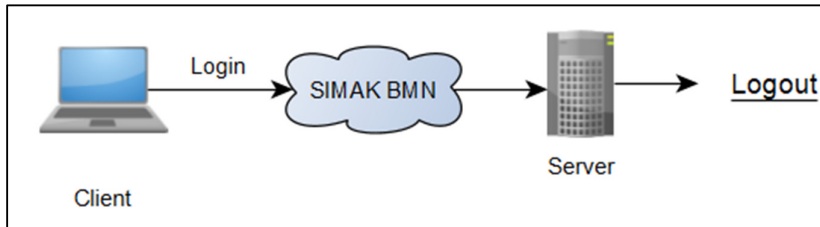
1. Saat ini BPS masih menggunakan sistem manual berbasis excel untuk mengontrol aset-aset yang dimiliki sehingga menyebabkan kesulitan dalam pengelolaan data-data aset.
2. Banyak masalah terkait keterangan pencatatan aset.
3. Kesulitan mengetahui riwayat aset di Kantor
4. Proses inventarisasi dinilai belum maksimal karena pencatatan terhadap aset masih kurang efektif.

3.3.2 Design Sistem

Tahap perancangan dan desain sistem ini adalah dengan melakukan desain sistem untuk memecahkan permasalahan yang terjadi di mana kita merancang sistem berorientasi objek menggunakan Flowchart sebagai gambaran struktur, use case diagram untuk menggambarkan perilaku pengguna terhadap aplikasi yang dibuat dan UML *Sequence Diagram* untuk memberikan gambaran visual tentang bagaimana objek-objek berkomunikasi dan berkolaborasi satu sama lain dalam scenario.

a) Gambaran Umum Sistem

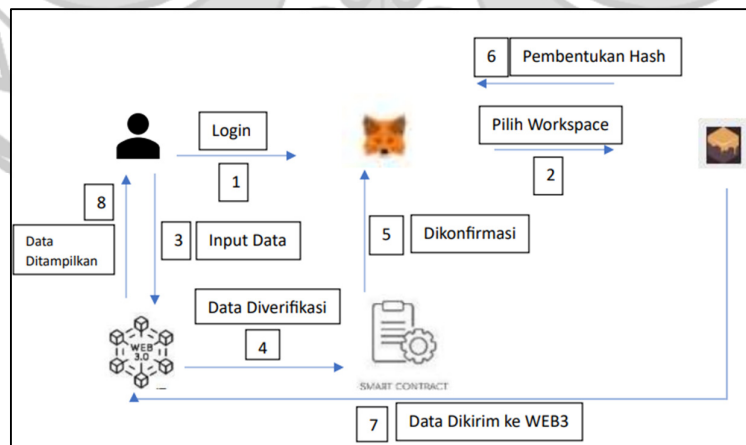
Gambaran traditional yang digunakan oleh *website* pada umumnya adalah *client-server network*. Pada kasus ini semua data disimpan pada satu tempat agar mudah untuk diupdate, karena servernya bersifat sentralisasi. Artinya database yang dikontrol oleh beberapa admin dengan masing – masing permission. Seperti pada gambar 3.3 dibawah ini:



Gambar 3.3 Gambaran Umum Database Tradisional

Gambaran sistem yang diterapkan BPS yaitu masih menggunakan aplikasi SIMAK BMN pada proses inventarisasi asetnya dimana langkah yang dilakukan yaitu terlebih dahulu user login ke aplikasi SIMAK BMN dengan menggunakan password dan user name. Setelah login, user dapat menggunakan berbagai macam fitur didalam aplikasi tersebut dimana semua data yang dikelola didalam aplikasi tersebut tersimpan didalam server. Setelah selesai menggunakan aplikasi SIMAK BMN, user diharuskan logout agar informasi tidak disalahgunakan oleh pihak yang tidak berwenang.

Untuk membangun sistem dengan menggunakan metode jaringan *Blockchain*, berikut dapat dilihat gambar 3.4 untuk Gambaran Sistemnya:



Gambar 3.4 Gambaran Umum Sistem *Blockchain*

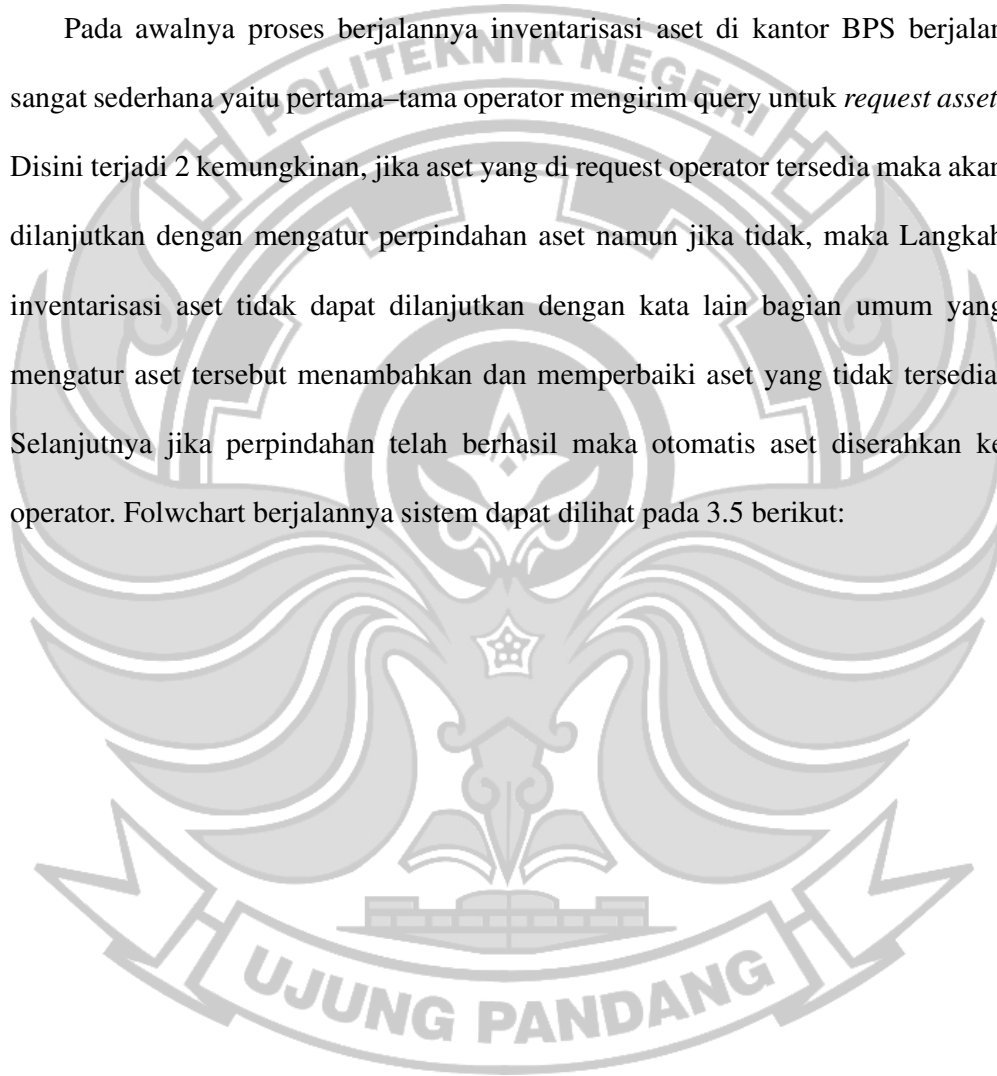
Pada penelitian ini dibangun suatu aplikasi Web3. Pada aplikasi web ini akan berjalan sesuai langkah – langkah berikut:

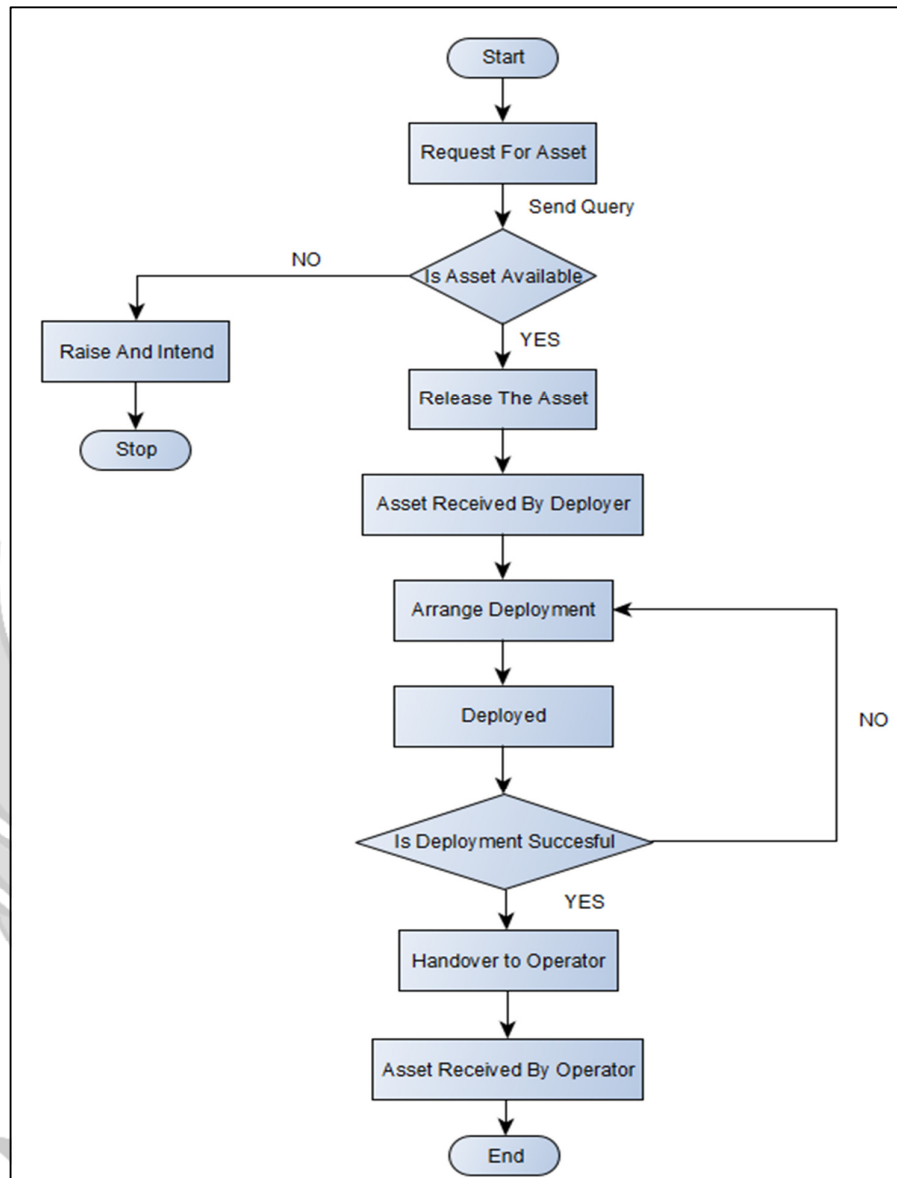
1. Masing – masing Laptop ter-*Install* aplikasi, system operasi, dan system web yang sama.
2. Pada saat membangun Aplikasi *Web* dilakukan pembuatan *Smart Contract* menggunakan bahasa pemrograman *Solidity*.
3. Aplikasi Web3 berfungsi sebagai aplikasi yang menjalankan program dan sebagai Interface dari aplikasi.
4. Pertama – tama admin login menggunakan akun MetaMask
5. Setelah berhasil login, admin selanjutnya diarahkan ke Ganache untuk memilih Workspace yang di pakai.
6. Setelah Workspace dipilih, admin melakukan input data ke dalam Web3 yang kemudian data tersebut di verifikasi didalam *Smart Contract* dan dikonfirmasi di MetaMask saat pengurangan gas untuk transaksi.
7. Jika proses tersebut sudah berhasil dikonfirmasi di MetaMask, selanjutnya terjadi pembentukan hash dan blok di Ganache.
8. Setelah transaksi sudah di validasi smart contract dan transaksi berhasil di konfirmasi MetaMask dan Ganache, data aset tersebut akan disimpan di dalam *Blockchain* yang selanjutnya akan dikembalikan ke aplikasi Web3 untuk ditampilkan kepada admin.

b) Flowchart

Flowchart disini berfungsi untuk menjelaskan atau menggambarkan setiap proses yang terjadi dalam sistem. Ini akan lebih memudahkan bagi peneliti untuk menjelaskan progres pengembangan software yang sedang digarapnya.

Pada awalnya proses berjalannya inventarisasi aset di kantor BPS berjalan sangat sederhana yaitu pertama-tama operator mengirim query untuk *request asset*. Disini terjadi 2 kemungkinan, jika aset yang di request operator tersedia maka akan dilanjutkan dengan mengatur perpindahan aset namun jika tidak, maka Langkah inventarisasi aset tidak dapat dilanjutkan dengan kata lain bagian umum yang mengatur aset tersebut menambahkan dan memperbaiki aset yang tidak tersedia. Selanjutnya jika perpindahan telah berhasil maka otomatis aset diserahkan ke operator. Flowchart berjalannya sistem dapat dilihat pada 3.5 berikut:

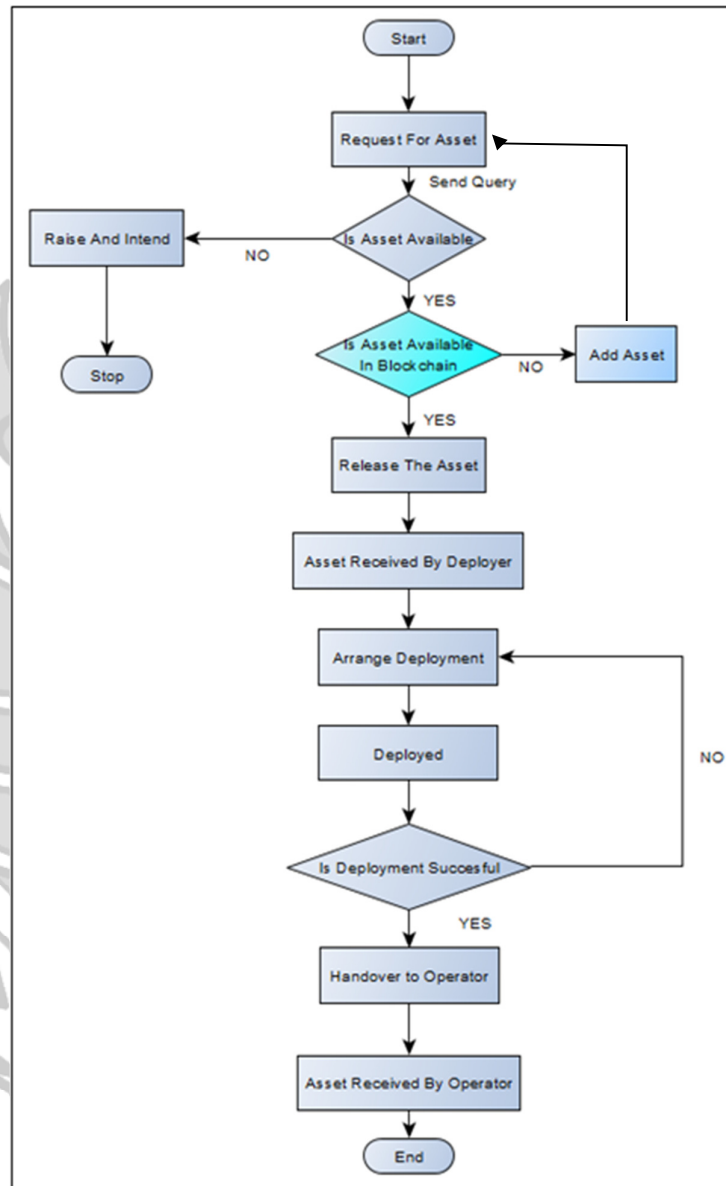




Gambar 3.5 Flowchart yang diterapkan BPS

Berikut gambaran sistem yang disarankan untuk memperbaiki sistem yang sementara diterapkan dapat dilihat pada gambar 3.5 inventarisasi aset dengan *Blockchain* pertama – tama mengumpulkan data pada aset kemudian disimpan pada platform didalam *Blockchain*. Hal ini memungkinkan pelacakan pergerakan aset dan kepemilikan aset secara aman dan transparan. Teknologi *Blockchain* juga dapat

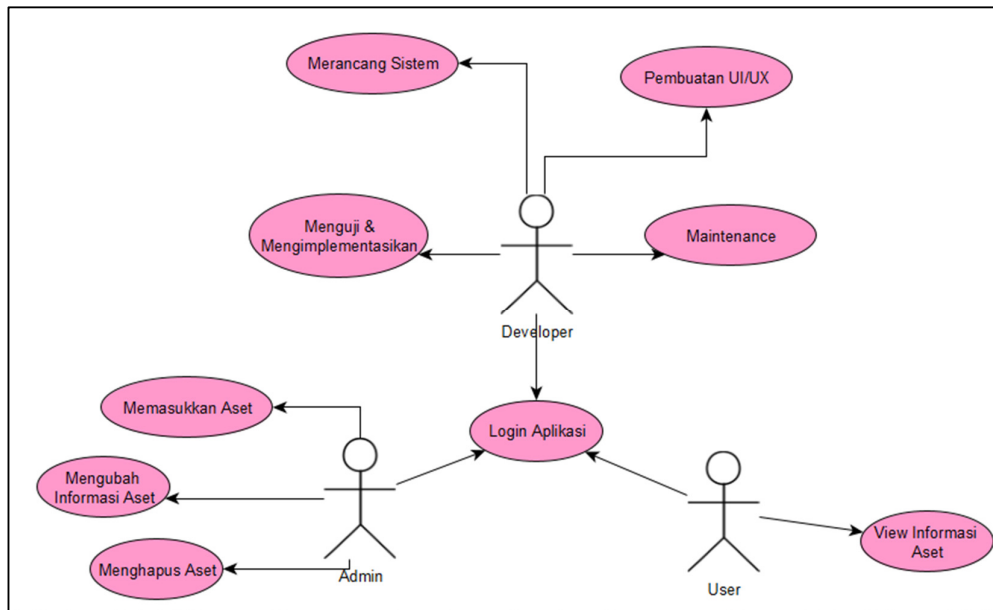
memfasilitasi pelacakan dalam rantai pasokan yang membantu manajemen aset dan keberlanjutan.



Gambar 3.6 Flowchart yang disarankan

Sumber: (Mehta, 2021)

c) *Use Case Diagram*



Gambar 3.7 Use Case Diagram

Gambar 3.7 Menunjukkan *Use Case Diagram* yang berfungsi membantu dalam mengidentifikasi siapa saja yang berinteraksi dalam a dan langkah apa saja yang dapat diambil. Pada penelitian ini terdapat 3 aktor utama yaitu *Developer*, Admin (Bagian Umum) dan User.

Developer memiliki peran sebagai berikut:

- a. Merancang aplikasi. *Developer* merancang arsitektur dan desain aplikasi, menentukan bagaimana komponen-komponen aplikasi akan berinteraksi satu sama lain dan menciptakan rencana implementasi.
- b. Membuat UI/UX. Proses ini merupakan proses dimana *Developer* merancang dan mengembangkan antarmuka pengguna dalam hal ini aplikasi yang *Support* dengan *Blockchain* yaitu WEB3.

- c. Pengujian dan Implementasi. Pengujian yang dilakukan *Developer* melibatkan pemeriksaan terhadap aplikasi seperti identifikasi *bug*, kesalahan atau masalah potensial lainnya sebelum aplikasi digunakan oleh *End User*.
- d. Maintenance. *Developer* bertanggung jawab untuk memantau kinerja aplikasi dan merespons masalah atau *bug* yang muncul.

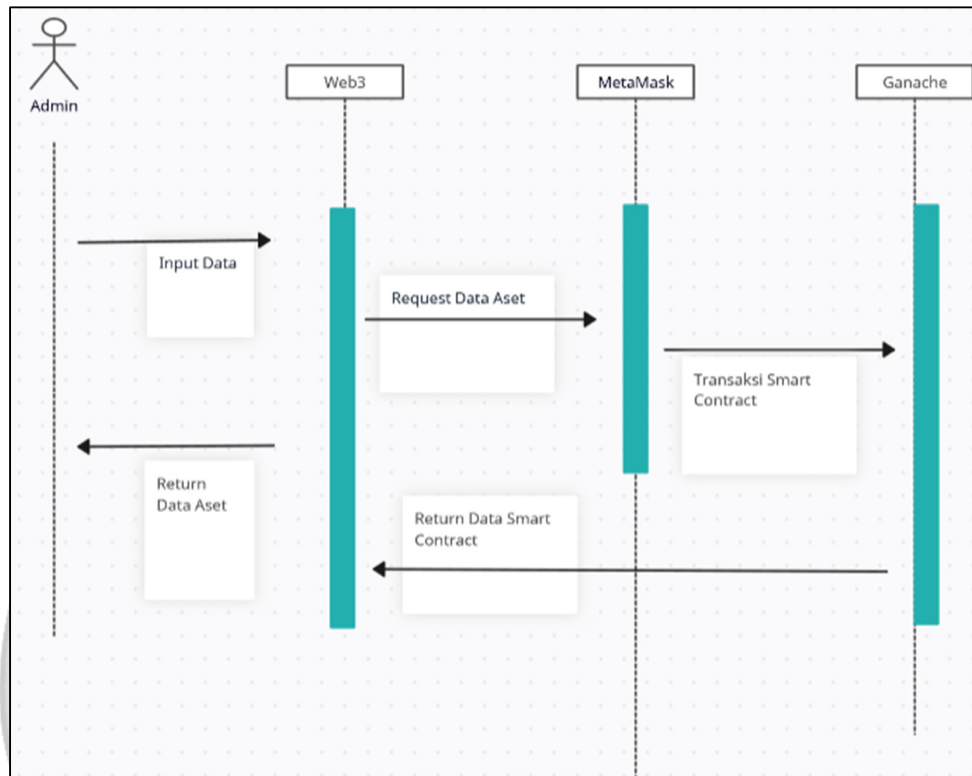
Admin (Bagian Umum) memiliki peran yaitu dapat melakukan CRUD didalam aplikasi yang telah dibuat. *Create*, admin dapat membuat data baru didalam aplikasi. *Read*, admin dapat membaca dan melihat data secara detail yang telah ada didalam aplikasi. *Update*, admin dapat memperbarui data serta mengedit data yang telah ada didalam aplikasi. Selanjutnya hak akses terakhir yang bisa dilakukan oleh admin yaitu *Delete*, yang dimana admin dapat menghapus data yang telah diinputkan didalam aplikasi.

Sedangkan *User* hanya diberikan hak akses yaitu melihat informasi aset yang telah diinputkan oleh admin. Peran ini penting dalam menjaga keamanan dan integrasi data, menghindari manipulasi data dan memastikan bahwa informasi data tetap terlindungi dari akses user selain dari admin.

d) Rancangan UML *Sequence Diagram*

Model *sequence diagram* dibuat untuk menggambarkan bagaimana interaksi antara objek dalam suatu skenario atau proses. Tujuan utama dari Diagram Urutan adalah untuk memberikan gambaran visual tentang bagaimana objek-objek berkomunikasi dan berkolaborasi satu sama lain dalam skenario tertentu.

Visualisasi dari *sequence diagram* dapat dilihat pada Gambar 3.8



Gambar 3.8 *Sequence Diagram*

Pada gambar 3.8 tersebut memiliki 1 aktor dan objek yaitu Web3, MetaMask dan *Blockchain Ethereum* local yaitu Ganache. Proses pada *Sequence Diagram* tersebut diawal dengan admin melakukan input data dan setelah itu melakukan reques data akun pada MetaMask dan terjadilah transaksi *Smart Contract* ke *Blockchain Ethereum* local yakni Ganache dan hasilnya yang berupa data didalam *Smart Contract* dikembalikan ke Web3 untuk ditampilkan kembali ke admin.

3.3.3 Coding

Pengkodean pada perangkat lunak merupakan proses penulisan bahasa program agar perangkat lunak tersebut dapat dijalankan oleh mesin. Aplikasi ini

dibangun menggunakan bahasa pemrograman *solidity* dan untuk membangun aplikasi antarmuka pengguna (UI) dalam aplikasi ini menggunakan bahasa pemrograman *React.js*

Bahasa pemrograman *solidity* merupakan bahasa pemrograman tingkat tinggi yang menerapkan konsep OOP yang digunakan untuk membuat *Smart Contract*. *Smart Contract* yang dimaksud adalah program yang mengatur perilaku akun dalam *Blockchain* Ethereum. Peran *Solidity* dalam *Smart Contract* yaitu mengirim ether ke *Smart Contract* dan juga mengirim ether dari *Smart Contract* ke *Contract* lain atau alamat lain. Compiler untuk *Smart Contract* dengan *solidity* menggunakan *Remix-IDE (Online Browser)* dan *Microsoft Visual Studio Code* serta text editor lainnya.

3.3.4 Implementasi dan Pengujian

Pada tahap implementasi dan pengujian juga dilakukan dalam pembuatan aplikasi ini. Implementasi merupakan penerapan cara kerja aplikasi. Pengujian ialah cara atau teknik yang dilakukan memastikan apakah semua fungsi fitur aplikasi ini bekerja dengan baik.

Adapun metode pengujian pada penelitian ini, ialah menerapkan bentuk pengujian seperti pengujian fungsionalitas, dan skalabilitas. Pengujian aplikasi ini dilakukan dengan cara mengamati keluaran (Output) dari berbagai masukan dan membandingkan output aplikasi web yang dibangun dengan output yang dilihat oleh pengguna sehingga sesuai dengan tujuan yang akan dicapai pada penelitian ini.

a) Implementasi

1. Pengadaan perangkat

Untuk pengadaan perangkat yang dibutuhkan dalam membangun aplikasi web ini, yaitu dapat dilihat pada subbab 3.2 pada bagian perangkat keras (*Hardware*).

2. Instalasi dan Konfigurasi

Pada tahap ini melakukan proses instalasi perangkat lunak yang diperlukan dalam membuat aplikasi web. *Software* yang dibutuhkan dapat dilihat pada subbab 3.2 pada bagian perangkat lunak (*Software*).

3. Pembuatan Aplikasi

Tahap ini dilakukan proses pengkodean script untuk membangun aplikasi web, sampai akhirnya dapat berintegrasi dengan pengguna. Pada tahap ini dilakukan juga pembuatan user interface (UI) sebagai sarana demonstrasi, dan yang terakhir mengembangkan *Smart Contract* yang mengatur perilaku akun dalam *Blockchain* Ethereum.

- b) **Pengujian**

Tujuan dari pengujian ini adalah untuk mengetahui apakah perangkat lunak atau software yang telah kita bangun berfungsi seperti tujuan pada awalnya. Pengujian pada penelitian ini menerapkan 2 bentuk pengujian sebagai berikut:

1. Pengujian Fungsionalitas merupakan pengujian untuk memastikan bahwa berbagai fitur dan fungsi *Blockchain* dapat berjalan sesuai dengan yang diharapkan. Pada pengujian ini terdapat 2 jenis pengujian fungsionalitas yaitu, pengujian fungsionalitas pada Server dan pengujian fungsionalitas pada node lain yang terhubung dengan Ganache sebagai server.

2. Pengujian Skalabilitas. Pengujian Skalabilitas adalah pengujian untuk memastikan bahwa *Blockchain* dapat mengakomodasi jumlah transaksi yang besar didalam aplikasi *Blockchain*. Untuk pengujian skalabilitas aplikasi *Blockchain* dengan jaringan lokal dapat dilakukan dengan memantau kinerja aplikasi saat jumlah transaksi dan pengguna meningkat dan menganalisis untuk mengidentifikasi titik kegagalan dan degradasi kinerja.



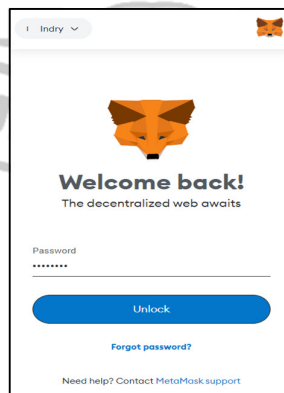
BAB IV HASIL DAN PEMBAHASAN

Pada penelitian ini dalam pembuatan Aplikasi Inventory Aset menggunakan Jaringan *Blockchain*, Hasil yang didapatkan yaitu terbentuknya *User Interface* (UI) sebagai sarana demonstrasi, berjalannya *Smart Contract* yang dapat mengatur perilaku akun di dalam jaringan *Blockchain* dan menghubungkan antara MetaMask, Ganache dan Web3.

Hasil yang didapatkan dari penelitian ini berdasarkan dari rancangan, tahapan – tahapan pelaksanaan dan pengujian yang sesuai dengan pokok permasalahan dan ruang lingkup penelitian. Penelitian ini berfokus pada pembuatan Aplikasi Inventory Aset menggunakan Jaringan *Blokckchain* mulai dari pembuatan aplikasi Web3 menggunakan *React.js*, pembuatan *Smart Contract* menggunakan *Solidity* dan menghubungkan MetaMask dengan Ganache sebagai *Blockchain* lokal untuk membantu simulasi transaksi yang terjadi didalam aplikasi.

4.1 Hasil Antarmuka WEB3

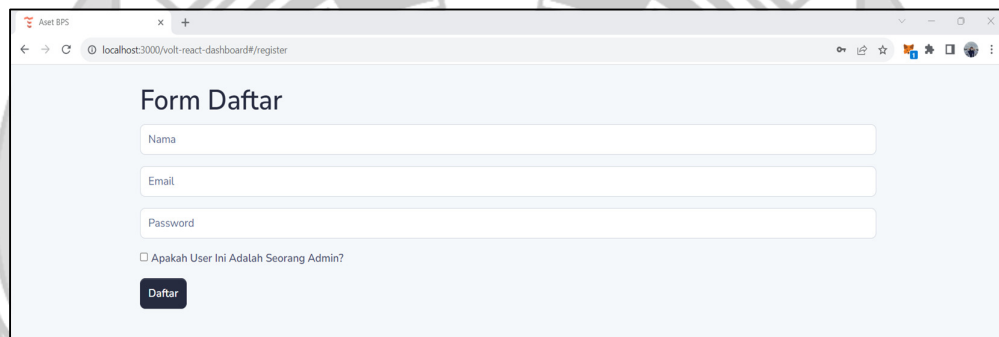
4.1.1 Login aplikasi menggunakan MetaMask



Gambar 4.1 Tampilan Login MetaMask

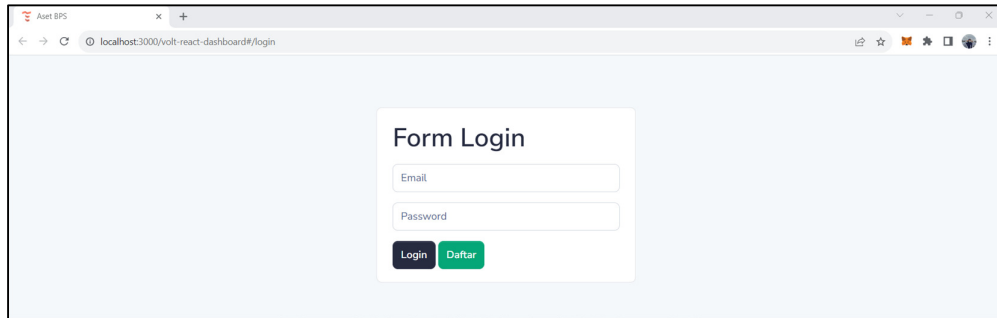
MetaMask berfungsi sebagai dompet kripto yang memungkinkan pengguna menyimpan, mengirim, dan menerima aset kripto seperti Ethereum serta berinteraksi dengan berbagai aplikasi desentralisasi (DApps). Untuk menggunakan MetaMask, pengguna diarahkan untuk *login* terlebih dahulu. Untuk *login*, pengguna diwajibkan mempunyai akun yang terhubung ke jaringan *Blockchain* yang tersedia.

4.1.2 Login dan Register Aplikasi Web3

A screenshot of a web browser window showing a registration form. The browser's address bar displays 'localhost:3000/volt-react-dashboard#/register'. The form is titled 'Form Daftar' and contains three input fields: 'Nama', 'Email', and 'Password'. Below these fields is a checkbox labeled 'Apakah User Ini Adalah Seorang Admin?'. At the bottom of the form is a dark button labeled 'Daftar'.

Gambar 4.2 Tampilan Register

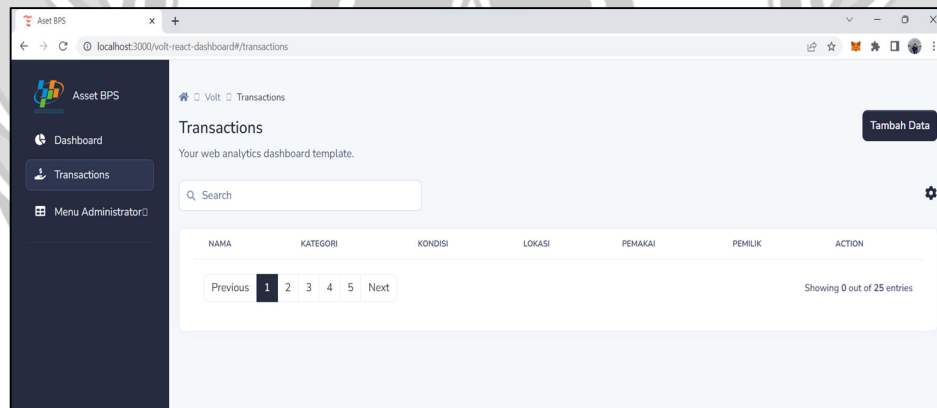
Pada Gambar 4.2 Tampilan Register di atas dapat dilihat bahwa pengguna baru dari aplikasi harus mengisi form daftar terlebih dahulu dan juga pengguna harus memiliki akun MetaMask karena pada proses transaksi yang berhubungan dengan *Smart Contract* harus di konfirmasi didalam MetaMask. Dan juga sebelumnya telah dijelaskan bahwa MetaMask merupakan jembatan antara Web3 dengan pengguna.



Gambar 4.3 Tampilan Login

Pada gambar 4.3 Tampilan Login diatas merupakan antarmuka yang memungkinkan pengguna untuk mengakses aplikasi dengan memasukkan informasi yang valid sesuai informasi yang sudah tercatat di akun MetaMask.

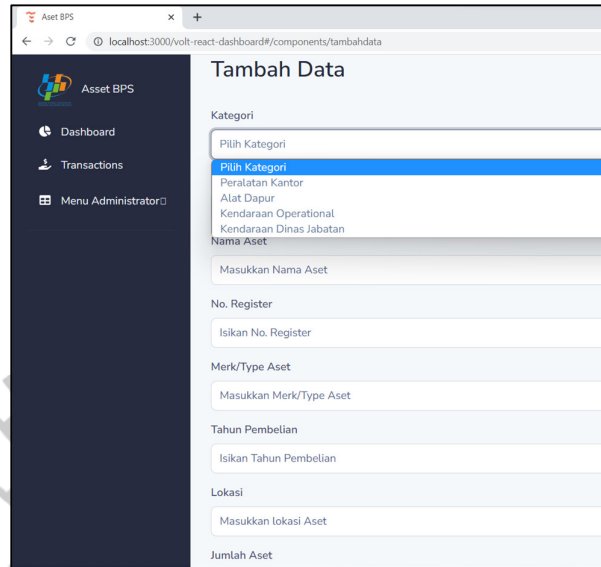
4.1.3 Halaman Index untuk admin



Gambar 4.4 Tampilan Halaman Index

Halaman index adalah halaman utama atau tampilan awal dari aplikasi desentralisasi (DApp). Halaman ini merupakan antarmuka pengguna pertama yang dilihat oleh pengguna saat mengakses DApp setelah login melalui MetaMask. Pada tampilan ini menampilkan informasi terkait aset BPS dan juga menampilkan fitur-fitur seperti Dashboard, Transactions, dan Menu Administrator.

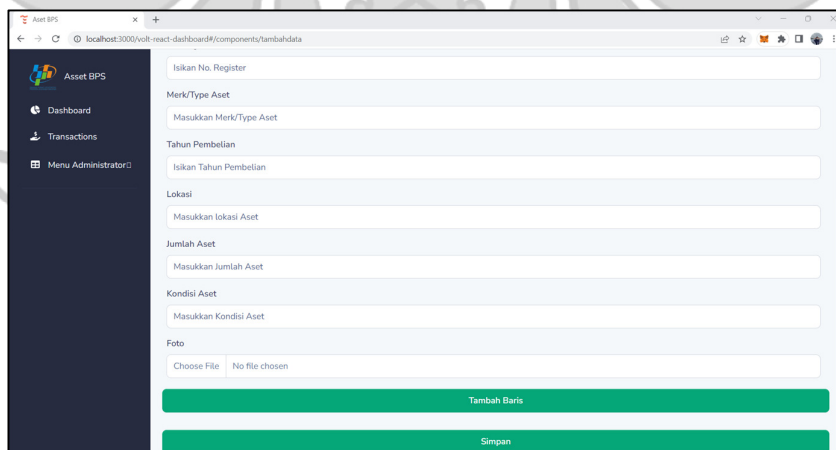
4.1.4 Halaman Tambah Data



Gambar 4.5 Tampilan Tambah Data Aset

Dapat dilihat pada gambar 4.5, di halaman ini terdapat *form* untuk mengisi data aset. Pada *form* data terdapat berbagai macam keterangan data aset yang harus diisi, setelah itu terdapat tombol “*simpan*” yang berfungsi sebagai perintah menyimpan data aset ke dalam *Blockchain* Ethereum.

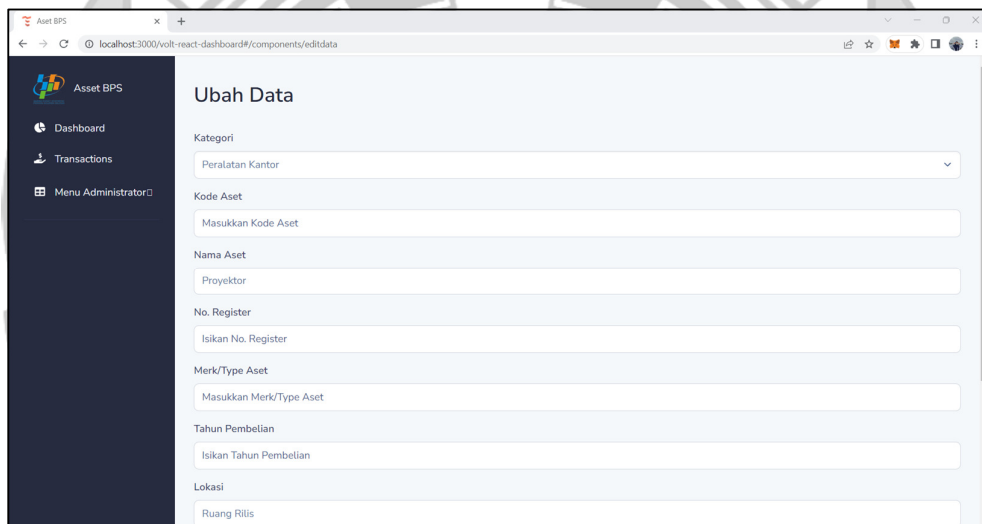
4.1.4 Halaman Tambah lebih dari satu data



Gambar 4.6 Halaman Tambah lebih dari satu data

Dapat dilihat pada gambar 4.6 halaman Tambah lebih dari satu data ini merupakan halaman yang dapat menambah form tambah data aset lebih dari satu data sehingga admin akan lebih dipermudah saat melakukan input data aset lebih dari satu, admin tidak perlu kembali ke halaman Transaksi untuk melakukan tambah data. Halaman ini dapat mengefisienkan waktu karna dapat menginput data sekaligus lebih dari satu data.

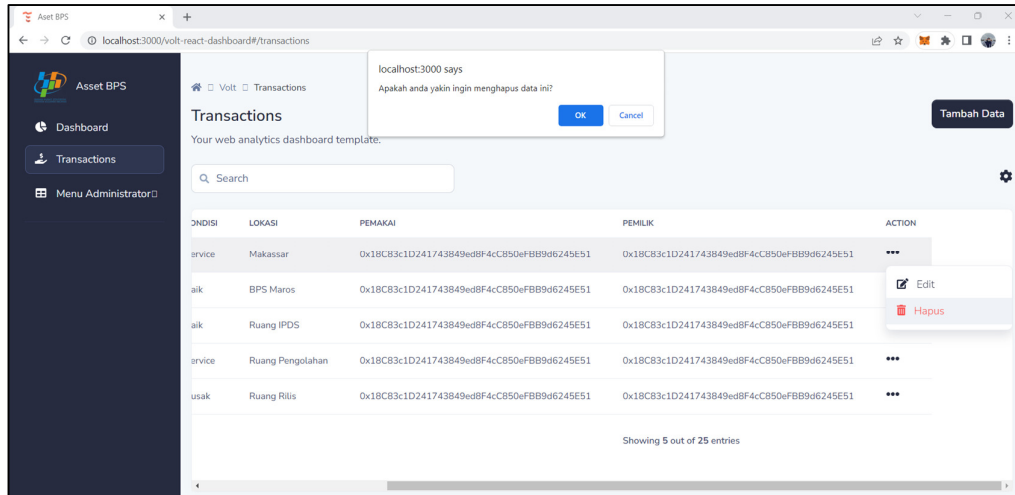
4.1.5 Halaman Edit Data



Gambar 4.7 Halaman Edit Data

Pada gambar 4.7 Halaman Edit Data merupakan halaman yang digunakan untuk mengubah data jika keterangan data berbeda dengan keterangan data sebelumnya. Misalnya jika sebuah aset yang sebelumnya memiliki keterangan kondisi baik, namun setelah digunakan aset tersebut rusak maka halaman ini yang digunakan untuk memperbarui data aset tersebut atau terdapat perubahan keterangan aset lainnya.

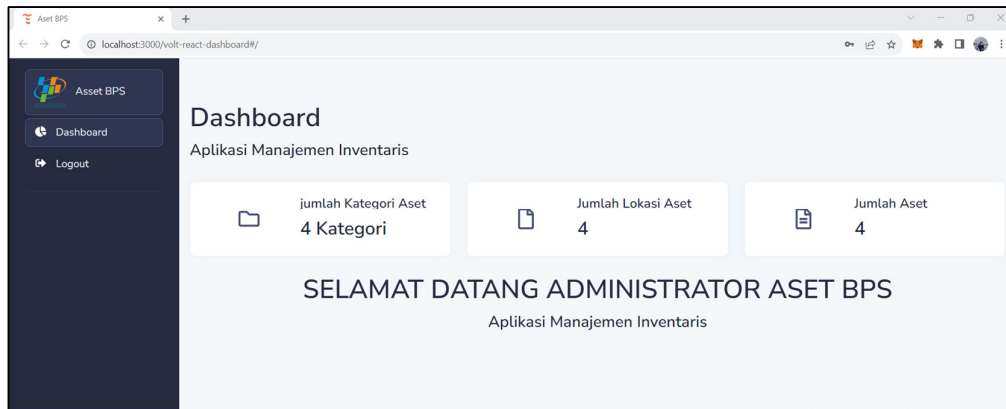
4.1.6 Halaman Hapus Data



Gambar 4.8 Halaman Hapus Data

Dapat dilihat pada Gambar 4.8 Halaman Hapus Data ini memfasilitasi penghapusan data aset terkait. Pengguna dalam hal ini admin perlu mengkonfirmasi tindakan penghapusan. Proses penghapusan melibatkan transaksi pada *Blockchain*, yang dapat mengharuskan pengguna membayar biaya transaksi dalam bentuk kripto. Data yang dihapus mungkin tidak benar-benar dihilangkan dari *Blockchain*, melainkan diberi status "dihapus" dalam *Smart Contract* yang mengelolanya untuk menjaga integritas *History*. Halaman ini memungkinkan pengguna untuk mengelola data dengan akurat dan sesuai dengan prinsip terdesentralisasi aplikasi Web3.

4.1.7 Halaman Index untuk User



Gambar 4.9 Halaman Index untuk User

Pada gambar 4.9 Halaman Index User merupakan halaman utama atau tampilan awal dari aplikasi desentralisasi (DApp) untuk user. Pada halaman ini merupakan halaman yang hanya bisa menampilkan informasi data berbeda dengan tampilan untuk admin yang bisa melakukan CRUD. Di halaman ini terdapat informasi mengenai aset yang dapat secara dinamis akan sesuai dengan informasi yang di ubah oleh admin.

4.2 Pembuatan *Smart Contract* dengan *Blockchain* lokal Ganache

Pengembangan smart contract Ethereum dilakukan pada *Blockchain* lokal, karena dengan menggunakan *Blockchain* lokal, pengujian dan pengkodean smart contract dapat dilakukan secara gratis dan lebih efisien. Pada penelitian ini menggunakan *Blockchain* lokal dari Ganache seperti pada Gambar 4.10.

MNEMONIC		HD PATH	
fog result result donor wire unknown resource blur example vote organ connect		m44'60'0'@account_index	
ADDRESS	BALANCE	TX COUNT	INDEX
0x18C83c1D241743849ed8F4cC850eFBB9d6245E51	99.98 ETH	34	0
ADDRESS	BALANCE	TX COUNT	INDEX
0xeDa86B0cb401545F995c05320A3326bA5B85b865	100.00 ETH	0	1
ADDRESS	BALANCE	TX COUNT	INDEX
0x6E4c70FDD6a65A7B681A7c020dd821329B0A35D1	100.00 ETH	0	2
ADDRESS	BALANCE	TX COUNT	INDEX
0x97bAFeca1b4307535e0E906d6436A288C2C4579B	100.00 ETH	0	3
ADDRESS	BALANCE	TX COUNT	INDEX
0x1e9b9D2853317543534906D7befa73a8F4FB8378	100.00 ETH	0	4
ADDRESS	BALANCE	TX COUNT	INDEX
0x7e88dD7C1BE6675B9eEF3E98e36DF103f8089892	100.00 ETH	0	5
ADDRESS	BALANCE	TX COUNT	INDEX
0xa8E7C1e897e2E8060f0edc147c894fF7c97598A4	100.00 ETH	0	6

Gambar 4. 10 *Blockchain* Lokal Ganache

Ganache bertujuan untuk menjalankan *Blockchain* versi localhost selayaknya *Blockchain* Ethereum pribadi, sehingga dapat digunakan untuk menjalankan pengujian, menjalankan perintah, dan memeriksa keadaan sambil mengontrol bagaimana rantai beroperasi. Setelah itu install framework *Truffle*. Framework *Truffle* berfungsi untuk mengintegrasikan pengkodean website dengan pengkodean *Smart Contract*, selain itu *Truffle* dapat melakukan pengujian *Smart Contract* menggunakan Terminal. Langkah selanjutnya membuat *Smart Contract* pada aplikasi ini terdapat 2 *Smart Contract* yaitu untuk proses inventarisasinya dan untuk proses login user. Berikut *Script Smart Contract* yang digunakan dalam aplikasi.

```

src > contracts > Inventaris.sol
1  pragma experimental ABIEncoderV2;
2  pragma solidity >=0.4.22 <0.9.0;
3
4  contract Inventaris {
5      uint totalInventori = 0;
6      struct Inventori {
7          string nama;
8          string kondisi;
9          string lokasi;
10         string kategori;
11         address pemakai;
12         address owner;
13     }
14     mapping(uint256 => Inventori) public listInventori;
15
16     function jumlahInventori() public view returns (uint) {
17         return totalInventori;
18     }
19     function ambilData(uint index) public view returns (Inventori memory) {
20         return listInventori[index];
21     }
22     function simpanData(
23         string memory _nama,
24         string memory _lokasi,
25         string memory _kondisi,
26         string memory _kategori
27     ) public {
28         require(bytes(_nama).length > 0, "Nama Tidak Boleh Kosong!");
29         address _owner = msg.sender;
30         totalInventori++;
31         listInventori[totalInventori] = Inventori(
32             _nama,
33             _kondisi,
34             _lokasi,
35             _kategori,
36             _owner,
37             _owner
38         );
39     }
40     function ubahData(
41         uint _index,
42         string memory _nama,
43         string memory _lokasi,
44         string memory _kondisi,
45         string memory _kategori
46     ) public {
47         require(bytes(_nama).length > 0, "Nama Tidak Boleh Kosong!");
48         address _pemakai = msg.sender;
49         address _owner = listInventori[_index].owner;
50         listInventori[_index] = Inventori(
51             _nama,
52             _kondisi,
53             _lokasi,
54             _kategori,
55             _pemakai,
56             _owner
57         );
58     }
59     function hapusData(uint _index) public {
60         if (_index >= totalInventori) return revert("Data tidak ditemukan");
61         totalInventori--;
62         return delete listInventori[_index];
63     }
64 }

```

Gambar 4. 11 Script Smart Contract untuk inventarisasi

Pada Gambar 4.11 yang merupakan Script Smart Contract diatas terdapat 2 method, yakni *get* dan *add*. Method *get* berfungsi untuk mengambil data dari

Blockchain. Sedangkan *Method add* berguna untuk menambahkan data dari transaksi kedalam *Blockchain*.

```
2 pragma experimental ABIEncoderV2;
3 pragma solidity >=0.4.22 <0.9.0;
4
5 contract Auth {
6     struct User {
7         address addr;
8         string name;
9         string email;
10        string password;
11        string role;
12    }
13
14    mapping(address => User) user;
15
16    // user registration function
17    function registerUser(
18        address _address,
19        string memory _name,
20        string memory _email,
21        string memory _password,
22        bool _isAdmin
23    ) public returns (bool) {
24        require(user[_address].addr != msg.sender);
25        user[_address].addr = _address;
26        user[_address].name = _name;
27        user[_address].email = _email;
28        user[_address].password = _password;
29        user[_address].role = _isAdmin ? "admin" : "user";
30        return true;
31    }
32
33    // user login function
34    function loginUser(
35        address _address,
36        string memory _email,
37        string memory _password
38    ) public view returns (bool) {
39        if (
40            keccak256(abi.encodePacked(user[_address].password)) ==
41            keccak256(abi.encodePacked(_password)) &&
42            keccak256(abi.encodePacked(user[_address].email)) ==
43            keccak256(abi.encodePacked(_email))
44        ) {
45            return true;
46        } else {
47            return false;
48        }
49    }
50
51    function checkRole(address _address) public view returns (string memory) {
52        return user[_address].role;
53    }
54 }
```

Gambar 4. 12 Script Smart Contract untuk login

Gambar 4.12 menjelaskan *Smart Contract* untuk proses registrasi dan login.

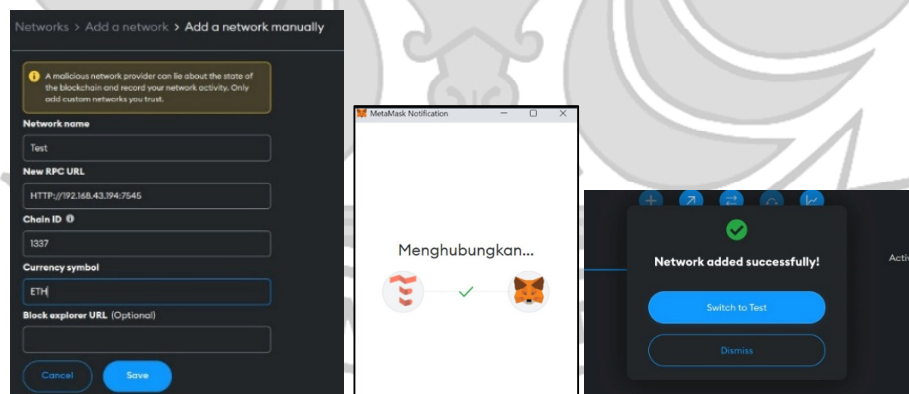
Pada *Smart Contract* ini terdapat 2 *Method* yaitu *Get* dan *Add*. Seperti yang telah dijelaskan sebelumnya bahwa metode *Get* berfungsi untuk mengambil data dari *Blockchain* pada saat proses login dilakukan sedangkan metode *Add* berfungsi

untuk menambah data kedalam *Blockchain* pada saat proses registrasi dilakukan user.

4.3 Menghubungkan MetaMask Pengguna Lain ke Ganache Server

Untuk menghasilkan pengembangan dan pengujian aplikasi yang lebih akurat, penting untuk menambahkan MetaMask pengguna lain ke Ganache Server dengan tujuan pengembangan kolaboratif, artinya semua pengguna dapat menguji dan berinteraksi dengan aplikasi yang sama dalam lingkungan yang konsisten. Tujuan lain yaitu Uji Skenario Multi-Pengguna dimana beberapa entitas atau akun berinteraksi satu sama lain dalam jaringan. Dengan demikian mereka dapat mensimulasikan dan menguji skenario aplikasi dengan lebih efisien. Berikut Langkah – Langkah cara menghubungkan MetaMask Pengguna Lain ke Ganache Server:

1. Memastikan pengguna lain yang akan dihubungkan sudah terinstal MetaMask
2. Mengkoneksikan MetaMask ke Ganache



Gambar 4.13 Konfirmasi *New Network*

Dalam *New Network* terdapat beberapa form yang perlu diisi yaitu *Network Name* atau nama jaringan baru, *New RPC URL* yang dimana merupakan RPC

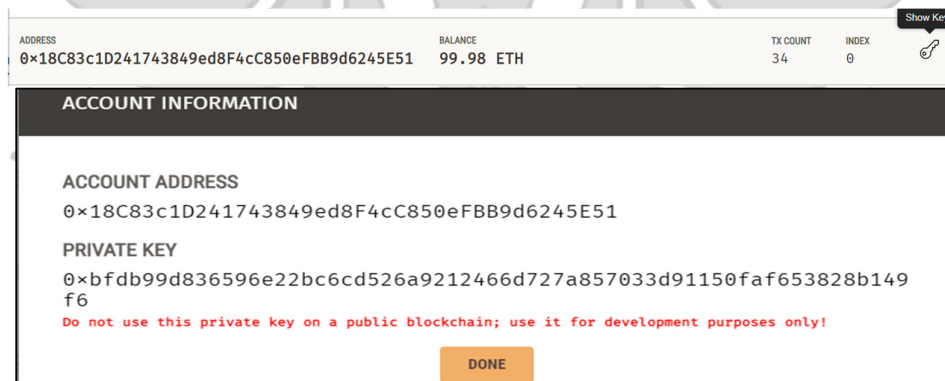
Server yang tentunya dilihat pada Ganache Server. Untuk Chain ID harus disesuaikan dengan Chain ID di Ganache atau yang biasa digunakan pada Ganache yaitu 1337. *Currency symbol* atau symbol mata uang yang diisi sesuai preferensi dalam hal ini menggunakan ETH.

3. Menyimpan konfigurasi dan memilih jaringan Ganache

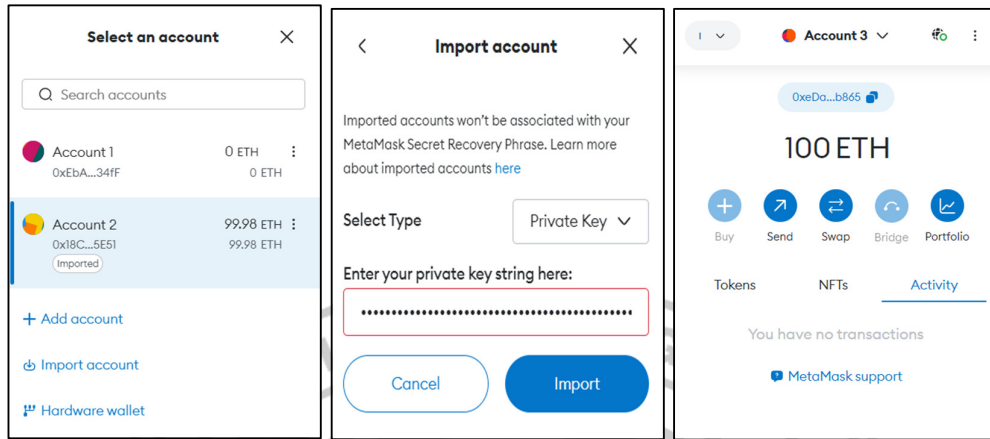
Setelah melakukan konfigurasi klik “Save” kemudian kembali ke layer utama MetaMask dengan menggunakan jaringan baru yang telah dibuat. Untuk memastikan MetaMask sudah terhubung ke Ganache dapat diketahui dengan melihat Alamat akun dan saldo awal yang sesuai dengan yang ada di Ganache.

4.4 Import Akun dari Ganache ke MetaMask

Untuk dapat melakukan simulasi transaksi di Web3, user harus memiliki akun *test network* berupa Ganache yang merupakan *Blokchain* lokal. Ketika mengimport akun Ganache ke MetaMask kita harus menggunakan *Private Key* yang terdapat didalam akun Ganache. Berikut tampilan *Private Key* pada Ganache:



Gambar 4.14 *Private Key* pada Akun Ganache



(1)

(2)

(3)

Gambar 4.15 Import Akun ke MetaMask

Gambar 4.15 menjelaskan bagaimana proses *Import Akun* dari Ganache ke MetaMask dimana terdapat Langkah-langkah sebagai berikut:

1. Login ke *Blockchain* lokal Ganache kemudian *Copy Private Key* dari salah satu akun di Ganache
2. Login ke ekstensi MetaMask, klik ikon profil atau gambar di kanan atas, pilih "*Import Account*", pilih opsi "*Private Key*", tempelkan *Private Key* akun Ganache yang telah salin kemudian klik "*Import*".

Akun dari Ganache sekarang akan ditambahkan ke MetaMask dan dapat digunakan untuk berinteraksi dengan aplikasi Web3.

4.5 Pengujian Fungsionalitas

Tabel 4. 1 Hasil Pengujian pada Node Server

No	Fitur Pengujian	Yang diharapkan	Pengamatan	Kesimpulan
1	Melakukan Registrasi	Berhasil registrasi dan masuk ke halaman login	Diarahkan masuk ke halaman login	Berhasil

No	Fitur Pengujian	Yang Diharapkan	Pengamatan	Kesimpulan
2	Melakukan Login dengan data yang benar	Masuk kehalaman <i>Dashboard</i>	Diarahkan langsung ke <i>Dashboard</i>	Berhasil
3	Melakukan Tambah data Aset	Data Aset berhasil ditambahkan	Beralih kehalaman tambah data	Berhasil
4	Melakukan tambah data lebih dari satu data	Data aset lebih dari satu data dapat ditambahkan	Beralih kehalaman tambah data dan muncul form untuk tambah data lebih dari satu	Berhasil
5	Melakukan pengeditan data	Menampilkan data yang sudah ada, lalu di edit	Kembali ke halaman transaksi dan muncul data baru yang sudah di edit	Berhasil
6	Menghapus data Aset	Aset yang sudah diputihkan dihapus datanya	Muncul alert penghapusan data dan kembali ke halaman transaksi	Berhasil

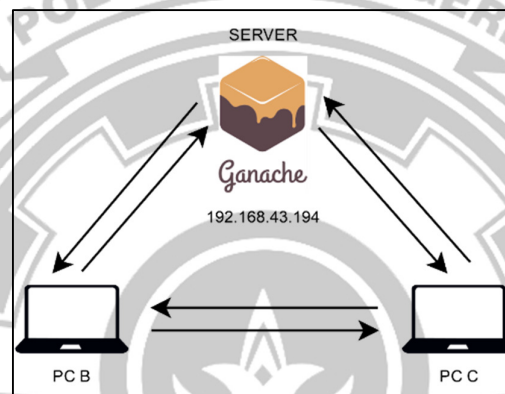
Tabel 4.2 Hasil Pengujian pada pengguna lain yang terhubung

No	Fitur Pengujian	Yang diharapkan	Pengamatan	Kesimpulan
1	Koneksi ke Server	Berhasil terhubung ke Ganache sebagai Server	Pengguna/node lain berhasil ditambahkan kedalam jaringan	Berhasil
2	Melakukan Registrasi	Berhasil registrasi dan masuk ke halaman login	Diarahkan masuk ke halaman login	Berhasil
3	Melakukan Login	Masuk kehalaman <i>Dashboard</i>	Diarahkan langsung ke <i>Dashboard</i>	Berhasil
4	Melakukan Tambah data Aset	Data Aset berhasil ditambahkan	Beralih kehalaman tambah data	Berhasil
5	Melakukan tambah data lebih dari satu data	Data aset lebih dari satu data dapat ditambahkan	Beralih kehalaman tambah data dan muncul form untuk tambah data lebih dari satu	Berhasil
6	Melakukan pengeditan data	Menampilkan data yang sudah ada, lalu di edit	Kembali ke halaman transaksi dan muncul data baru yang sudah di edit	Berhasil
7	Menghapus data Aset	Aset yang sudah diputihkan dihapus datanya	Muncul alert penghapusan data dan kembali ke halaman transaksi	Berhasil

Pada Tabel 4.1 dan 4.2 dapat dilihat hasil dari pengujian fungsionalitas aplikasi bahwa semua fitur Aplikasi Inventory Aset berjalan sesuai dengan yang diharapkan. Berdasarkan hasil pengamatan selama pengujian fungsional aplikasi baik pada node server dan node lain yang terhubung, semua fitur dalam aplikasi berhasil digunakan dan menjalankan fungsinya masing-masing.

4.6 Pengujian Skalabilitas

Pengujian Skalabilitas dilakukan dengan cara menghubungkan beberapa PC ke server dan masing-masing PC tersebut melakukan transaksi sesuai skenario pengujian. Pengujian ini dilakukan untuk melihat apakah jaringan *Blockchain* pada aplikasi ini mampu menangani lonjakan jumlah transaksi dengan baik atau tidak.



Gambar 4.16 Topologi

Tabel 4.3 Hasil pengujian Skalabilitas

Pengguna	Jumlah Transaksi	Durasi Waktu Transaksi (detik)	Waktu Respon Rata-rata (detik)
PC B PC C	1	6,87 detik 6,07 detik	6,47 detik
PC B PC C	5	39,2 detik 37,2 detik	7,64 detik
PC B PC C	10	81 detik 90 detik	8,55 detik
PC B PC C	20	156 detik 148 detik	7,6 detik
Pengguna	Jumlah Transaksi	Durasi Waktu Transaksi (detik)	Waktu Respon Rata-rata (detik)

PC B	30	237 detik	8,03 detik
PC C		245 detik	
PC B	40	332 detik	8,05 detik
PC C		312 detik	
PC B	50	446 detik	7,65 detik
PC C		319 detik	

Berdasarkan gambar 4.16 dapat dilihat skenario pengujian pada aplikasi, dimana PC B dan PC C harus terhubung ke server dalam hal ini menggunakan Ganache. Kedua PC harus terhubung dengan jaringan yang sama dengan server yaitu 192.168.43.194. kemudian untuk bisa mengakses Aplikasi Inventory Aset tersebut menggunakan port 3000.

Tabel 4.3 merupakan hasil dari pengujian skalabilitas dimana pengujian dilakukan dengan variasi jumlah inputan data atau transaksi yaitu 1, 5, 10, 20, 30, 40 dan 50 transaksi. Selanjutnya digunakan Durasi Waktu Transaksi dan Waktu Respon Rata-rata, dimana Durasi Waktu Transaksi merupakan jumlah semua waktu yang tercatat pada saat konfirmasi jumlah transaksi hingga masing-masing transaksi tercatat di Server. Sedangkan Waktu Respon Rata-rata merupakan Total Durasi Waktu Transaksi dibagi banyaknya Jumlah Transaksi.

Berdasarkan Hasil dari pengujian ini jaringan *Blockchain* mampu merespon Ketika jumlah transaksi meningkat. Pengujian ini membantu dalam mengevaluasi tingkat skalabilitas jaringan yakni mampu menangani lonjakan aktivitas dengan baik.

BAB V PENUTUP

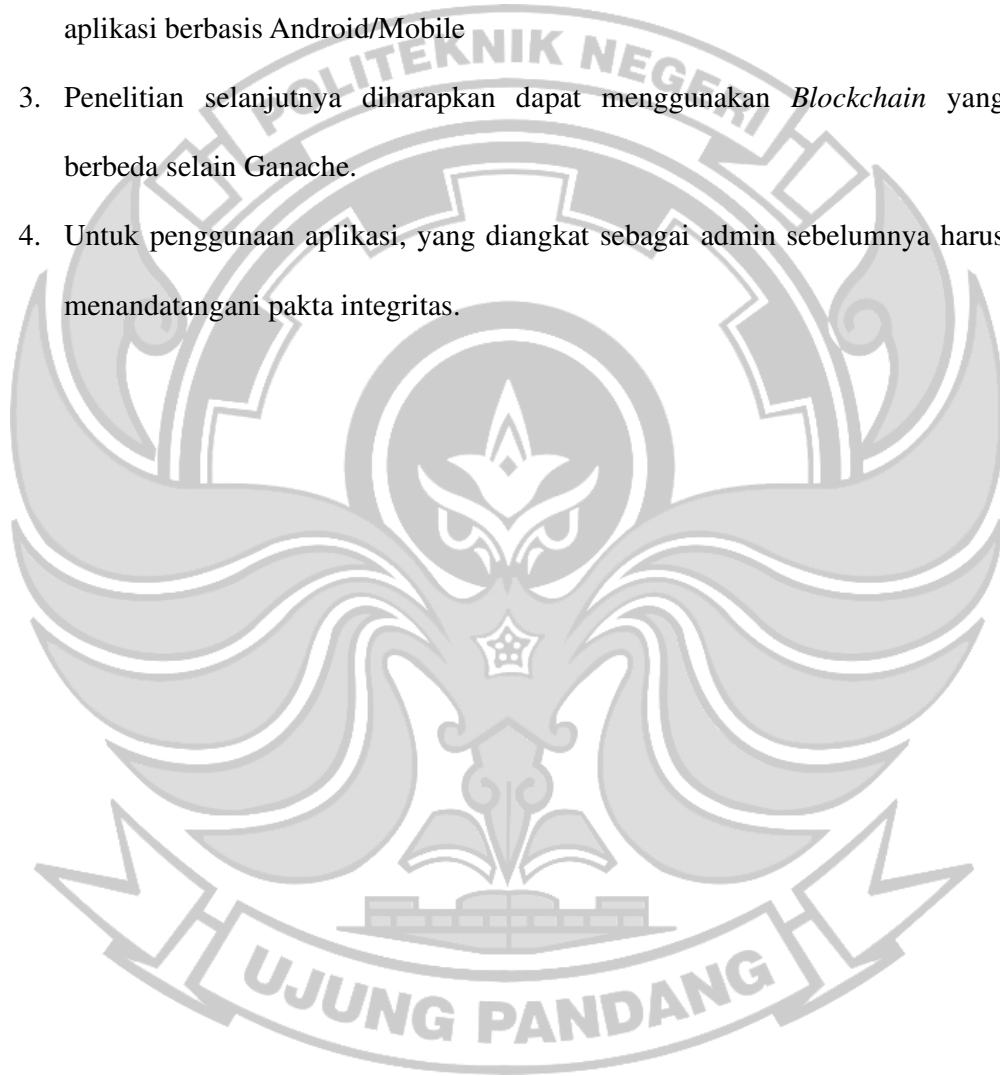
5.1 Kesimpulan

Berdasarkan hasil penelitian tentang Aplikasi Inventory Aset menggunakan Jaringan *Blockchain* dapat ditarik kesimpulan sebagai berikut. Penelitian ini berhasil membangun Aplikasi Inventory Aset menggunakan Jaringan *Blockchain* dengan tujuan untuk manajemen aset dan untuk menunjukkan beberapa keunggulan jaringan *Blockchain* dalam inventarisasi aset tersebut. Dalam proses pengembangan aplikasi ini membutuhkan Ganache yang berperan sebagai *Blockchain* lokal untuk pengujian dan simulasi *Smart Contract* serta menyimpan data transaksi yang terjadi didalam aplikasi. Selanjutnya digunakan extension MetaMask untuk transaksi sekaligus sebagai penghubung antara pengguna dengan Web3. Dari beberapa tahapan tersebut dilakukan pengujian fungsionalitas dan skalabilitas sesuai dengan skenario pengujian. Hasil dari pengujian fungsionalitas yaitu semua fitur pada aplikasi ini sudah berjalan dengan baik sesuai skenario pengujian yang diharapkan. Sedangkan hasil dari pengujian skalabilitas yaitu penggunaan jaringan *Blockchain* pada aplikasi ini mampu menangani jumlah transaksi yang semakin meningkat dengan hasil *respon time* yang diterima server hampir sama disetiap transaksi.

5.2 Saran

Adapun saran untuk pengembangan Aplikasi Inventory Aset menggunakan Jaringan *Blockchain* dimasa yang akan datang yaitu:

1. Penelitian selanjutnya diharapkan dapat mengembangkan *Smart Contract* yang dimana didalam aplikasi dapat mengunggah berbagai macam *Type File* seperti *.doc*, *.img*, dll.
2. Mengembangkan antarmuka aplikasi dari aplikasi berbasis web menjadi aplikasi berbasis Android/Mobile
3. Penelitian selanjutnya diharapkan dapat menggunakan *Blockchain* yang berbeda selain Ganache.
4. Untuk penggunaan aplikasi, yang diangkat sebagai admin sebelumnya harus menandatangani pakta integritas.



DAFTAR PUSTAKA

- Alexander Sugiharto, S. H., Muhammad Yusuf Musa, M. B. A., Falahuddin, M. J., & ST, M. B. A. (2022). *NFT & Metaverse: Blockchain, Dunia Virtual & Regulasi* (Vol. 1). Indonesian Legal Study for Crypto Asset and *Blockchain*.
- Aminah, S., Bhaskoro, S. B., & Sunarya, A. S. (2020). Penerapan Quick Response Code pada Sistem Digitalisasi Inventaris Laboratorium Berbasis Android. *JTERA (Jurnal Teknologi Rekayasa)*, 5(2), 207. <https://doi.org>
- Apriliasari, D., & Seno, B. A. P. (2022). Inovasi Pemanfaatan *Blockchain* dalam Meningkatkan Keamanan Kekayaan Intelektual Pendidikan. *Jurnal MENTARI: Manajemen, Pendidikan Dan Teknologi Informasi*, 1(1), 68–76.
- Bhosale, K., Akbarabbas, K., Deepak, J., & Sankhe, A. (2019). *Blockchain based Secure Data Storage*. 5058–5061.
- BPS. (2021). *Badan Pusat Statistik Provinsi Sulawesi Selatan*. Badan Pusat Statistik. <https://sulsel.bps.go.id>
- Dewanto, A. (2021). *Potensi Maksimal Dari Harga Value Pada Sebuah Transaksi Yang Menggunakan Teknologi Blockchain Di Jaringan Ethereum*. Univeristas Komputer Indonesia. <https://elibrary.unikom.ac.id>
- Fadhillah, Y., Samosir, K., Angriawan, R., Jamaludin, J., Ardiana, D. P. Y., Parewe, A. M. A. K., Yuswardi, Y., Simarmata, J., Pakpahan, A. F., & Multazam, M. T. (2022). *Teknologi Blockchain dan Implementasinya*.
- Fidiana, F., Kautsar, I. A., & Maika, M. R. (2018). *Ekonomi Digital*. Umsida Press.
- Liu, X., Barenji, A. V., Li, Z., Montreuil, B., & Huang, G. Q. (2021). *Blockchain-based smart tracking and tracing platform for drug supply chain*. *Computers & Industrial Engineering*, 161, 107669.
- Marpaung, J. (2018). Pengaruh Penggunaan Gadget Dalam Kehidupan. *KOPASTA: Jurnal Program Studi Bimbingan Konseling*, 5(2), 55–64. <https://doi.org>
- Maulana, J. F. (2020). *Analisis Pelaksanaan Program Kampung Keluarga Berencana Dalam Meningkatkan Kesejahteraan Keluarga Di Kelurahan Air Dingin Kecamatan Bukit Raya Kota Pekanbaru (Studi Pada Kelompok Kegiatan Bina Keluarga Balita)*. Universitas Islam Riau.
- Mehta, J., Mehta, D., Jain, J., & Dholay, S. (2021, August). Asset tracking system using blockchain. In *2021 Asian Conference on Innovation in Technology (ASIANCON)* (pp. 1-7).
- Nasrudin, E. (2015). Efektivitas Sistem Informasi Manajemen Dan Akuntansi Barang Milik Negara (Simak-Bmn) Terhadap Pengelolaan Aset Negara. *Jurnal Akuntansi Universitas Jember*, 13(2), 45. <https://doi.org>

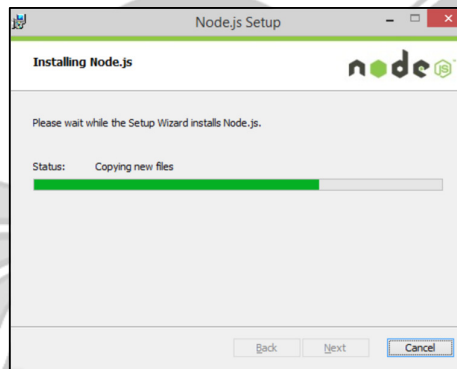
- Noorsanti, R. C., Yulianton, H., & Hadiono, K. (2018). *Blockchain - Teknologi Mata Uang Cryptocurrency. Prosiding Sendi_U 2018*, 978–979.
- Novendri, M. S., Saputra, A., & Firman, C. E. (2019). Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan Php Dan Mysql. *Lentera Dumai*, 10(2).
- Nugraha, A. C. (2022). Penerapan Teknologi *Blockchain* dalam Lingkungan Pendidikan. *Produktif: Jurnal Ilmiah Pendidikan Teknologi Informasi*, 4(1), 302–307. <https://doi.org>
- Pambudi, G. S., & Arvianto, A. (2016). *Berbasis Web Untuk Optimalisasi Penelusuran Aset Di Teknik Industri Undip*. XI(3), 187–196.
- Ratnawati, C. (2015). *Aplikasi Sistem Informasi Manajemen Dan Akuntansi Barang Milik Negara (Simak-Bmn) Berupa Aset Tetap Pada Kantor Wilayah Direktorat Jenderal Kekayaan Negara Jawa Timur*. Stie Perbanas Surabaya.
- Saputro, H. J., & Chandra, T. Y. (2021). Urgensi Pemulihan Kerugian Keuangan Negara Melalui Tindakan Pemblokiran Dan Perampasan Asset Sebagai Strategi Penegakan Hukum Korupsi. *Mizan: Journal of Islamic Law*, 5(2), 273–290.
- Shell, A. (2016). Perancangan Aplikasi E-Voting Dengan Sistem Smart Contract Berbasis Teknologi *Blockchain*. *Perancangan Aplikasi E-Voting Dengan Sistem Smart Contract Berbasis Teknologi Blockchain*, 7(2), 1–23. <http://www.joi.isoss.net>
- Supriyono, S. (2019). Penerapan ISO 9126 Dalam Pengujian Kualitas Perangkat Lunak pada E-book. *Matics*, 11(1), 9. <https://doi.org>
- Sutopo, A. H. (2023). *Pemrograman Blockchain Smart Contract di Polygon*. Topazart.
- Yusup, M., Aini, Q., Apriani, D., & Nursaputri, P. (2019). Pemanfaatan Teknologi *Blockchain* Pada Program Sertifikasi Dosen. *SENSITif: Seminar Nasional Sistem Informasi Dan Teknologi Informasi*, 365–371.

LAMPIRAN

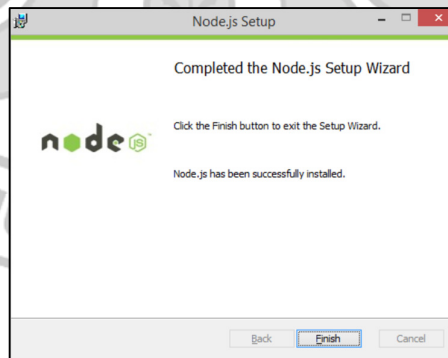
Lampiran 1. Proses Instalasi Tools

a. Proses Instalasi Node.JS

- 1) Unduh Node.JS *Installer* melalui website resminya
<https://node.js.org/download>
- 2) Jalankan *file* msi yang sudah diunduh
- 3) Ikuti petunjuk dan *User Agreement* Dan tunggu hingga proses instalasi selesai



- 4) Klik "Finish" untuk menyelesaikan proses instalasi



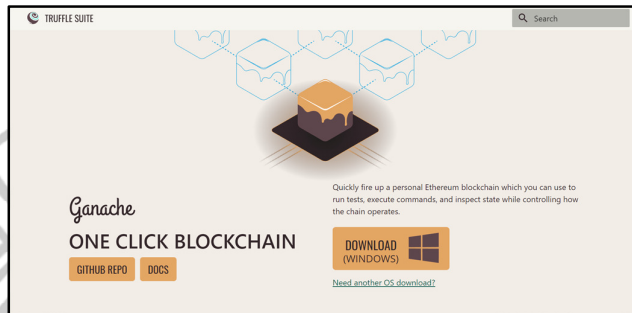
- 5) Proses Instalasi Node.JS telah berhasil

```
C:\Users\ACER>node -v  
v18.15.0
```


b. Proses Instalasi Ganache

4. Unduh *Installer* Ganache melalui situs resminya di

<https://trufflesuite.com/Ganache>

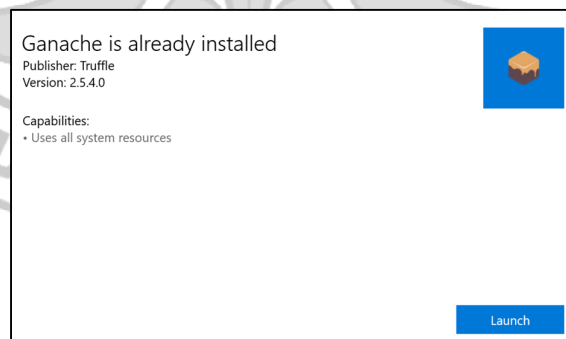


5. Setelah itu, buka *Installer* yang sudah diunduh dimana *Installer* tersebut berada dalam format appx.

 Ganache-2.5.4-win-x64

6. Ketika *file* tersebut dibuka, pengguna akan diarahkan ke *windows store* dan proses instalasi akan berjalan

7. Ketika proses instalasi sudah selesai maka akan muncul tampilan seperti berikut



8. Klik "Launch" dan proses instalasi selesai.

c. Proses Instalasi Ektensi MetaMask di Google Chrome

- 1) Buka Google Chrome
- 2) Kunjungi Chrome Web Store: Pergi ke Chrome Web Store dengan mengunjungi tautan berikut: <https://chrome.google.com/webstore>
- 3) Cari MetaMask: Di bilah pencarian Chrome Web Store, ketik "MetaMask" dan tekan tombol "Enter" atau klik ikon pencarian.
- 4) Pilih Ekstensi MetaMask
- 5) Instal Ekstensi: Setelah Anda memilih ekstensi MetaMask, klik tombol "Tambahkan ke Chrome" yang terletak di sebelah kanan atas jendela ekstensi. Sebuah jendela konfirmasi akan muncul.
- 6) Konfirmasi Instalasi: Ketika jendela konfirmasi muncul, klik tombol "Tambahkan ekstensi" untuk mengonfirmasi dan memulai proses instalasi.
- 7) Tunggu Hingga Selesai. setelah selesai diinstal, Anda akan melihat ikon MetaMask muncul di bilah alat Chrome di sebelah kanan atas.
- 8) Mulai MetaMask: Klik ikon MetaMask untuk membuka ekstensi. Anda akan diminta untuk membuat akun atau mengimpor akun yang sudah ada.
- 9) Setelah mengatur akun MetaMask, Anda siap untuk menggunakan ekstensi dan berinteraksi dengan aplikasi desentralisasi dan jaringan *Blockchain* yang didukung.