

Analisis dan Implementasi Sistem Pendeteksi Ijazah dan Transkrip Palsu dengan Menggunakan IPFS dan *Smart Contract Blockchain*

Muhammad Danil Muis¹, Parman Sukarno², Aulia Arif Wardana³

^{1,2,3} Universitas Telkom, Bandung

¹daniltelkom@students.telkomuniversity.ac.id, ²psukarno@telkomuniversity.ac.id,

³aulwardana@telkomuniversity.ac.id

Abstrak

Pemalsuan ijazah/transkrip merupakan salah satu masalah dalam dunia pendidikan. Dalam menyikapi hal ini, Lembaga Layanan Pendidikan Tinggi membuat Sistem Verifikasi Ijazah Secara Online (SIVIL) dan kebijakan Penomoran Ijazah Nasional (PIN). Sistem yang dibangun masih menggunakan penyimpanan secara terpusat sehingga rentan terhadap peretasan seperti SQLInjection yang membahayakan data. Oleh karena itu, sistem yang dikembangkan pada penelitian ini menggunakan penyimpanan terdistribusi untuk mencegah terjadinya serangan tersebut. Sistem ini menggunakan *InterPlanetary File System* (IPFS) untuk menyimpan data secara terdistribusi dan *smart contract blockchain* untuk menyimpan *hash file* ijazah/transkrip. Dalam mengetahui performansi sistem, maka dilakukan pengujian *Quality of Service* (QoS) dengan menggunakan parameter *throughput*, *packet loss* dan *delay* serta analisis penggunaan *Central Processing Unit* (CPU) dan *Random Access Memory* (RAM). Berdasarkan penelitian yang sudah dilakukan, sistem pendeteksi ijazah/transkrip palsu dapat dijalankan dengan baik dengan menggunakan 1 *node* sampai dengan 5 *node*. Nilai *throughput* terbaik pada saat proses pembuatan serta proses validasi ijazah/transkrip adalah dengan menggunakan 1 *node*. Nilai *packet loss* pada proses pembuatan serta proses validasi ijazah/transkrip memiliki kategori sangat bagus. Nilai *delay* pada proses pembuatan serta proses validasi ijazah/transkrip memiliki kategori sangat bagus.

Kata kunci : IPFS, Blockchain, Ijazah

Abstract

Falsification of diplomas/transcripts is one of the problems in education. In response to this, Lembaga Layanan Pendidikan Tinggi created a Sistem Verifikasi Ijazah Secara Online (SIVIL) and a Penomoran Ijazah Nasional Numbering (PIN) policy. The system still uses centralized storage, so it is vulnerable to attacks such as SQL injection, which endanger data. Therefore, the system developed in this research uses distributed storage to prevent these attacks. This system uses *InterPlanetary File System* (IPFS) to store data in a distributed manner and *Smart Contracts Blockchain* to store the diploma/transcript hashes. In knowing the system performance, a *Quality of Service* (QoS) test was carried out using *throughput*, *packet loss*, and *delay* parameters as well as analysis of the usage of *Central Processing Unit* (CPU) and *Random Access Memory* (RAM). Based on the research that has been done, the fake diploma/transcript detection system can be run properly by using 1 *node* to 5 *nodes*. The best *throughput* value during the process of making and validating the diploma/transcript is to use 1 *node*. The value of *packet loss* in the process of making and validating the certificate/transcript has a very good category. The value of *delay* in the process of making and validating the diploma/transcript has a very good category.

Keywords: IPFS, Blockchain, Diplomas

1. Pendahuluan

1.1. Latar Belakang

Menurut KBBI, ijazah merupakan surat izin yang diberikan sebagai tanda tamat belajar [1]. Pada tahun 2018, berdasarkan investigasi yang dilakukan oleh Tirto terdapat 873 ijazah yang diproduksi oleh kampus swasta di Tangerang. Namun, mahasiswa yang lulus hanya 145 orang, sehingga terdapat 728 ijazah palsu yang diterbitkan [2]. Dalam menyikapi permasalahan tersebut, Lembaga Layanan Pendidikan Tinggi (LL Dikti) menggagas kebijakan Penomoran Ijazah Nasional (PIN) sejak tahun 2017 [3]. Solusi lain yang diterapkan oleh LL Dikti adalah Sistem Verifikasi Ijazah Secara Online (SIVIL) yang terintegrasi dengan Pangkalan Data Pendidikan Tinggi (PD Dikti) secara online [3]. Sayangnya sistem yang dibangun masih menggunakan penyimpanan secara terpusat sehingga rentan terhadap peretasan yang dapat mengakibatkan perubahan data, kerusakan bahkan kehilangan data. Selain itu, penggunaan ijazah dan transkrip berbentuk cetak juga memiliki kelemahan seperti kerusakan dan kehilangan dokumen.

Pada penelitian ini akan membahas tentang analisis dan implementasi sistem pendeteksi ijazah dan transkrip palsu dengan menggunakan IPFS dan *smart contract blockchain*. *InterPlanetary File System* (IPFS) merupakan *distributed file system* yang bersifat *Peer to Peer* (P2P) dengan tujuan untuk menggantikan HTTP

(*Hypertext Transfer Protocol*). Selain IPFS, penelitian ini juga menggunakan *Blockchain* dan *smart contract* yang terintegrasi. Hal ini akan memberikan keuntungan dimana transaksi *peer to peer* dan *database* dapat dikelola secara aman dan terpercaya [4]. Penelitian ini juga akan melakukan pengujian performansi *Quality of Service* (QoS) pada sistem pendeteksi ijazah dan transkrip palsu dengan IPFS dan *smart contract Blockchain* serta analisis penggunaan *Central Processing Unit* (CPU) dan *Random Access Memory* (RAM). Analisis QoS akan menghasilkan waktu yang dibutuhkan dalam proses pengiriman data ke penerima (*throughput*), lama waktu yang dibutuhkan pada proses pengiriman (*delay*) dan banyaknya paket yang hilang dalam proses transmisi (*packet loss*).

1.2. Topik dan Batasannya

Pemalsuan ijazah dan transkrip masih marak dilakukan. Dalam mengatasi hal tersebut Lembaga Layanan Pendidikan Tinggi (LL Dikti) meluncurkan Penomoran Ijazah Nasional (PIN) dan Sistem Verifikasi Ijazah Secara Online (SIVIL). Solusi ini masih menimbulkan masalah lain dimana sistem penyimpanan terpusat rentan terhadap serangan SQLInjection. Dengan demikian rumusan masalah yang diangkat dalam penelitian ini adalah membangun sistem pendeteksi ijazah dan transkrip palsu dengan penyimpanan terdistribusi menggunakan IPFS dan *smart contract Blockchain*. Arsitektur terdistribusi ini terdiri dari beberapa *nodes*. Oleh karena itu, dibutuhkan pengujian *Quality of Service* (QoS) serta analisis penggunaan CPU dan RAM untuk mengetahui pengaruh jumlah *nodes* pada performansi sistem.

1.3. Tujuan

Tujuan dari penelitian ini adalah membangun sistem pendeteksi ijazah dan transkrip palsu dengan menggunakan IPFS dan *smart contract Blockchain*, mengetahui performa *Quality of Service* (QoS) pada sistem pendeteksi ijazah dan transkrip palsu dengan menggunakan IPFS dan *smart contract Blockchain*, mengetahui pengaruh jumlah *node* pada *cluster* terhadap performa *Quality of Service* (QoS) dan penggunaan CPU serta RAM pada sistem pendeteksi ijazah/transkrip palsu dengan menggunakan IPFS dan *smart contract Blockchain*.

1.4. Organisasi Tulisan

Buku tugas akhir ini disusun menjadi lima bagian yaitu Bab 1-Pendahuluan yang berisikan latar belakang, topik dan batasannya, dan tujuan penelitian. Bab 2-Studi terkait berisi studi literatur atau penelitian yang sudah dilakukan sebelumnya yang merupakan acuan untuk penelitian ini serta dasar teori penelitian. Bab 3-Perancangan sistem yang memaparkan alur sistem yang dilakukan penelitian ini. Bab 4-Implementasi dan Evaluasi, menjelaskan tentang pengimplementasian sistem dan analisis hasil penelitian. Bab 5-Penutup yang berisi kesimpulan penelitian dan saran untuk penelitian selanjutnya.

2. Studi Terkait

2.1. Penelitian Terkait

Pada penelitian yang akan dilakukan oleh peneliti berkaitan dengan implementasi sistem untuk memverifikasi ijazah dan transkrip serta penggunaan teknologi IPFS dan *smart contract Blockchain* untuk penyimpanan *file*. Penelitian ini memiliki perbedaan dibandingkan penelitian sebelumnya. Adapun perbedaan tersebut dari segi masalah yang diangkat, metode yang digunakan serta teknologi yang diimplementasikan. Berikut ini beberapa penelitian sebelumnya yang berkaitan dengan penelitian yang akan diangkat oleh peneliti.

Pada penelitian dengan judul “Analisis dan Implementasi Komunikasi Antar *Node* IPFS (Interplanary *File* System) Pada Smart Contract Ethereum” mengimplemmentasikan sistem berbasis web untuk menyimpan *file*. Penelitian ini menggunakan metode Network Development Life Cycle (NDLC) untuk membangun sistem. Hasil dari penelitian ini adalah aplikasi web terdistribusi dengan menggunakan IPFS dan *smart contract* untuk menyimpan *file*. Penelitian ini memiliki kelebihan dimana pengujian *Quality of Service* (QoS) menggunakan kapasitas *file* yang bervariasi. Namun, penelitian ini tidak membahas berapa banyak gas yang digunakan untuk setiap transaksi serta pengujian QoS tidak dijelaskan metodenya [9].

Pada penelitian dengan judul “Blockchain-Based, Decentralized Access Control for IPFS” membahas tentang implementasi aplikasi untuk menambahkan *file* dengan informasi sensitif dan membatasi akses ke *file* tersebut. Penelitian ini menggunakan Acl-IPFS untuk meningkatkan keamanan. Kelebihan dari penelitian ini adalah keamanan lebih terjamin. Namun, penggunaan Acl-IPFS memberikan perbedaan kecepatan yang signifikan dalam penyimpanan ke IPFS. Selain itu, pengujian juga hanya dilakukan dengan tiga kapasitas *file* yang berbeda [10].

Pada penelitian dengan judul “Validasi Ijazah dengan Menggunakan Watermarking dan QR Code pada Fakultas Teknik UNIS Tangerang” membahas tentang implementasi sistem yang digunakan untuk melakukan validasi ijazah. Sistem ini menggunakan watermark dan QR code dan pengujian sistem menggunakan metode

kuisioner. Kelebihan dari penelitian ini adalah menggunakan dua pengamanan yaitu watermark dan QR code. Namun, aplikasi yang dirancang masih berbasis desktop dan sistem tidak terdistribusi [11].

Pada penelitian dengan judul "Implementasi Tanda Tangan Digital Menggunakan RSA dan SHA-512 Pada Proses Legalisasi Ijazah" yang membahas tentang sistem verifikasi ijazah dengan digital signature menggunakan RSA dan SHA-512. Metode yang digunakan untuk membangun sistem ini adalah waterfall. Kelebihan penelitian ini adalah algoritma yang diterapkan memiliki hasil uji yang aman. Namun, sistem yang dibangun tidak terdistribusi dan dokumen tanda tangan digital yang disisipkan sangat banyak [12].

2.2. Dasar Teori

2.2.1. Ijazah

Menurut KBBI, Ijazah adalah surat izin yang diberikan sebagai tanda tamat belajar [2]. Menurut peraturan Menteri Pendidikan dan Kebudayaan Nomor 81 tahun 2004, Ijazah adalah dokumen pengakuan prestasi belajar dan/atau penyelesaian suatu jenjang pendidikan tinggi setelah lulus ujian yang diselenggarakan oleh perguruan tinggi [13]. Ijazah dapat diperoleh seseorang jika sudah menyelesaikan pendidikan formal, seperti Sekolah Dasar (SD) selama enam tahun, Sekolah Menengah Pertama (SMP) selama tiga tahun, Sekolah Menengah Atas (SMA) selama tiga tahun bahkan perguruan tinggi. Pentingnya kepemilikan ijazah didasari sebagai bukti yang digunakan untuk beberapa urusan penting terutama jabatan pekerjaan.

2.2.2. Penomoran Ijazah Nasional (PIN) dan Sistem Verifikasi Ijazah Secara Elektronik (SIVIL)

Penomoran Ijazah Nasional (PIN) merupakan proses penomoran ijazah nasional dengan menggunakan aplikasi yang disebut PIN. Nomor ijazah nasional ini akan dicantumkan pada ijazah yang diterbitkan oleh perguruan tinggi. Proses ini memiliki dua tahap yaitu reservasi atau booking nomor ijazah untuk calon lulusan dan pemasangan nomor ijazah dengan Nomor Induk Mahasiswa (NIM) calon lulusan. Nomor Ijazah Nasional ini terdiri dari 15 digit yang meliputi kode prodi, tahun lulusan, nomor urut dan check digit. Sistem Verifikasi Ijazah Secara Elektronik (SIVIL) merupakan aplikasi yang digunakan untuk memverifikasi nomor ijazah yang pernah diterbitkan dan memverifikasi keabsahan serta pengesahan nomor ijazah nasional [14].

2.2.3. Blockchain

Blockchain merupakan teknologi dasar dari Bitcoin yang sangat populer. *Blockchain* adalah sejenis struktur rantai yang mengkombinasikan blok data dalam urutan waktu dan menyerupai *ledger* terdistribusi atau *database*. Keamanan *Blockchain* dijamin oleh kriptografi asimetris dan tidak dapat dirusak ataupun dipalsukan [6]. *Blockchain* menggunakan mekanisme konsensus. Setiap transaksi yang menyebabkan perubahan pada *ledger Blockchain* akan ditandai, diverifikasi dan divalidasi oleh *nodes* yang menyimpan duplikasi dari *ledger* [15]. Berdasarkan kelebihan dan infrastruktur terdesentralisasi, teknologi *Blockchain* digunakan untuk permasalahan yang berkaitan dengan kepercayaan, efisiensi, privasi dan data *sharing* [15]. Hal ini menyebabkan teknologi *Blockchain* banyak digunakan pada seluruh sektor industri seperti keuangan, kesehatan, logistik, dll.

2.2.4. Ethereum

Ethereum merupakan *platform* terbuka yang digunakan untuk membangun aplikasi terdesentralisasi di atas *Blockchain* [17]. Semua komputasi yang diprogram pada Ethereum dikenakan biaya dan ditentukan dalam unit gas [18]. Gas adalah satuan ukuran yang digunakan untuk menghitung jumlah pekerjaan yang diselesaikan untuk suatu operasi [19]. Ethereum menggunakan Ether (ETH) digunakan sebagai *cryptocurrency* untuk melakukan pembayaran atas transaksi yang dilakukan pada *Blockchain* Ethereum. Setiap partisipan pada Ethereum dapat diidentifikasi melalui Ethereum Address (EA) [15].

Ethereum menggunakan *Proof of Work* (POW) sebagai protokol konsensus. Model keamanan yang diberikan oleh POW dilengkapi dengan digital signatures dan ditambahkan ke *ledger* atau *database* untuk memastikan bahwa *history* tidak bisa dirusak dan tidak bisa dipalsukan [18]. Ethereum *Blockchain* mengekspansi fungsionalitasnya dengan mengimplementasikan *smart contract*.

2.2.5. Smart Contract

Smart Contract adalah perwujudan dari kontrak atau perjanjian dalam dunia maya. Kontrak ini berbentuk sebuah kode dimana dapat tercipta apabila seluruh syarat pembuatan kontrak sudah terpenuhi [20]. Misalnya, ada transaksi antara penjual dan pembeli. Dalam dunia maya biasanya penjual akan meletakkan barang dagangannya di suatu *marketplace* agar ada yang bisa membeli. Kemudian, pembeli yang tertarik dapat langsung membeli barang tersebut. Ketika uang sudah berhasil diterima, *smart contract* akan terbentuk antara pembeli dan penjual. Kemudian, masing - masing pihak mendapatkan barangnya. Biasanya barang yang diperjual belikan adalah *cryptocurrency*.

2.2.6. Interplanary File System (IPFS)

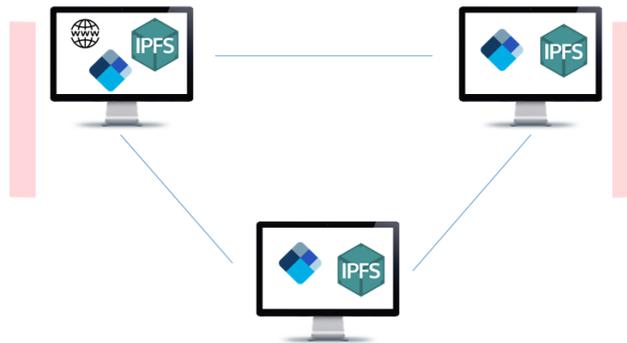
Interplanary File System (IPFS) merupakan *file system* terdistribusi yang bekerja dengan cara membuat seluruh *node* yang terhubung saling berbagi atau menggunakan *file system* yang sama [21]. *File* yang disimpan akan dipecah menjadi blok-blok kemudian didistribusikan ke *node* yang terhubung dengan IPFS [22]. IPFS dibuat untuk mengatasi sistem penyimpanan yang masih terpusat sehingga membutuhkan banyak *storage*. IPFS juga bisa memastikan bahwa hanya ada 1 *file* yang ada dalam *cluster*, sehingga tidak terjadi duplikasi dan lebih hemat *storage*.

2.2.7. Quality of Service (QoS)

Quality of Service (QoS) adalah kemampuan dari sebuah jaringan untuk menyediakan layanan yang lebih baik lagi bagi trafik yang melewatinya. QoS dapat diukur dengan menggunakan beberapa parameter, yaitu *throughput*, *packet loss*, dan *delay*. *Throughput* merupakan kecepatan (*rate transfer*) dalam pengiriman paket data dalam satuan kbps (kilo bit *per second*). *Throughput* memiliki jumlah total paket yang sukses diterima selama *interval* waktu tertentu dan dibagi dengan *interval* waktu tersebut. *Packet loss* merupakan paket yang hilang pada proses transmisi ke tujuannya. *Packet loss* dapat terjadi karena beberapa faktor seperti *overflow traffic* atau tabrakan dalam jaringan, *error* pada media fisik, *interface* atau *router* down dan kegagalan yang terjadi pada sisi penerima seperti *overflow* pada *buffer*. *Delay* merupakan waktu tunggu yang dibutuhkan untuk proses pengiriman dari *transmitter* sampai diterima oleh *receiver*.

3. Sistem yang Dibangun

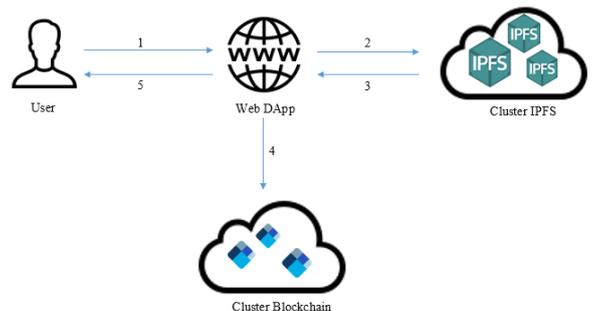
3.1. Perancangan Sistem



Gambar 1 Arsitektur Sistem yang Akan Dibangun

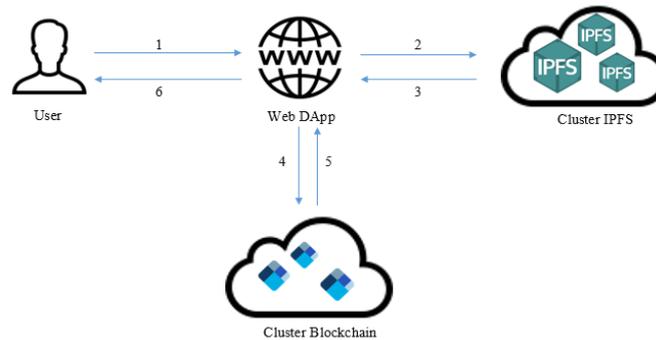
Sistem dirancang dengan menggunakan 5 *node*. Penggunaan jumlah *node* dimaksudkan untuk membuktikan sistem terdistribusi dan kaitannya dengan kualitas layanan yang akan dihasilkan sistem. *Node* yang berisi web DApp akan di-*install* IPFS dan akan membuat *cluster* IPFS. *Node* ini juga akan di-*install* Geth untuk membuat *Blockchain*. *Node* yang lain akan di-*install* IPFS dan bergabung dengan *cluster* yang sudah dibuat. Pada sistem yang akan dibangun tidak menggunakan Metamask sehingga penyimpanan dilakukan secara otomatis. Hal ini bertujuan untuk memberikan kemudahan bagi *user*.

3.2. Alur Sistem



Gambar 2 Alur Sistem Pembuatan Ijazah/Transkrip

Pada proses pembuatan ijazah/transkrip, terdapat beberapa tahapan pada (1) *user* melakukan *input form* pada web DApp lalu web DApp akan membuat *file* ijazah/transkrip berdasarkan data yang dimasukkan. Selanjutnya (2) ijazah/transkrip yang sudah dibuat akan dikirimkan ke *cluster* IPFS dan mengubah *file* ijazah/transkrip menjadi blok-blok untuk didistribusikan ke *node* yang berada pada *cluster*. Selanjutnya (3) *hash* dari *file* akan dikirimkan kembali ke web DApp. Selanjutnya (4) web DApp akan mengirimkan *hash* tersebut ke *cluster Blockchain* lalu *hash* tersebut akan didistribusikan ke *node* yang berada pada *cluster*. Terakhir (5) web DApp akan mengirimkan *file* ijazah/transkrip ke *user* untuk diunduh.



Gambar 3 Alur Sistem Validasi Ijazah/Transkrip

Pada Proses validasi ijazah/transkrip, terdapat beberapa tahapan pada (1) *user* mengunggah *file* ijazah/transkrip ke web DApp lalu (2) web DApp akan mengirimkan *file* tersebut ke *cluster* IPFS untuk diubah menjadi *hash*. Selanjutnya (3) *hash* akan dikirimkan kembali ke web DApp. Selanjutnya (4) web DApp akan mengirimkan *hash* untuk divalidasi oleh *cluster Blockchain*. Selanjutnya (5) hasil dari validasi akan dikirimkan kembali ke web DApp dan (5) hasilnya akan ditampilkan ke *user*.

3.3. Skenario Pengujian

Pengujian dilakukan dengan *user* membuat sebuah ijazah/transkrip pada web DApp sampai *user* menerima *file* ijazah/transkrip untuk diunduh. Selain itu juga dilakukan pengujian ketika *user* mengunggah ijazah/transkrip ke web DApp lalu divalidasi oleh *Blockchain* dan web DApp mengirimkan hasilnya ke *user*. Pengujian dilakukan dengan menambah jumlah *node* secara bertahap dari 1 sampai 5 *node*. Untuk setiap jumlah *node* akan dilakukan pembuatan ijazah/transkrip dan validasi ijazah/transkrip masing – masing sebanyak 10 kali percobaan dimana setiap percobaan dilakukan *request* sebanyak 10 kali. Tujuan dari pengujian tersebut adalah untuk mengetahui *Quality of Service* (QoS), penggunaan CPU, dan pemakaian RAM dari sistem yang sudah dibangun. Adapun spesifikasi *server* yang digunakan dalam penelitian ini adalah sebagai berikut :

Tabel 1. Spesifikasi Server

Komponen	Detail
<i>Processor</i>	Intel(R) Xeon(R) Bronze 3204 CPU @ 1.90GHz
RAM	16 GB
<i>Operating System</i>	Windows Server 2019 Datacenter
<i>Harddisk</i>	DELL PERC H330 SCSI 1TB

Server ini akan dibuat virtual *node* sebanyak 5 *node* yang memiliki spesifikasi sebagai berikut :

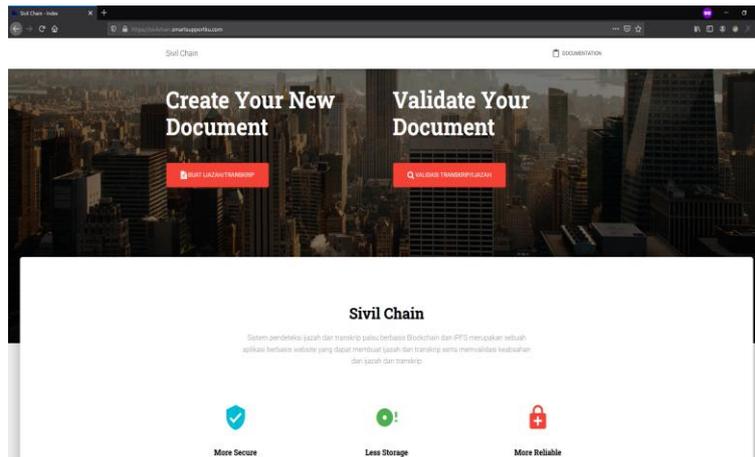
Tabel 2. Spesifikasi Node

Komponen	Detail
<i>Processor</i>	1 <i>core</i>
RAM	2048 MB
<i>Operating System</i>	Debian 10.8.0 AMD 64
<i>Harddisk</i>	48 GB

4. Evaluasi

4.1. Aplikasi Pendeteksi Ijazah/Transkrip Palsu

Aplikasi yang dikembangkan oleh peneliti merupakan aplikasi pembuatan dan pengecekan ijazah/transkrip palsu dengan menggunakan IPFS dan *Smart Contract Blockchain*. Adapun tampilan awal dari aplikasi dapat dilihat pada **Gambar 4**.

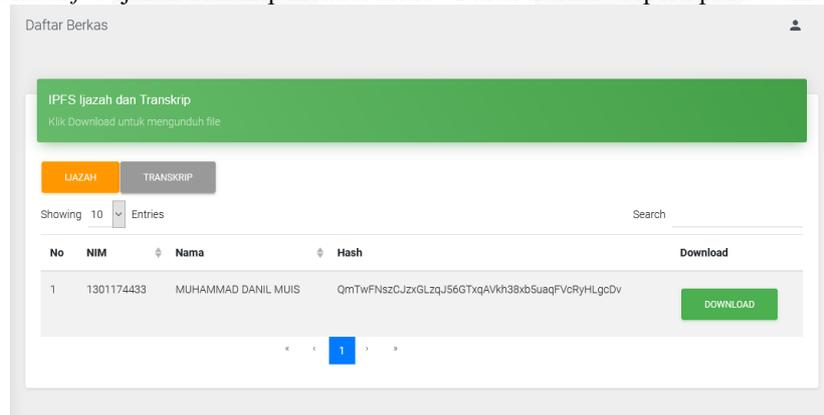


Gambar 4 Halaman Awal Aplikasi Pendeteksi Ijazah/Transkrip Palsu

Menu pertama dalam aplikasi ini adalah pembuatan ijazah/transkrip. Dalam membuat ijazah/transkrip maka *user* diwajibkan untuk mengisi data pada *form* seperti pada **Gambar 5**.

Gambar 5 Form Pembuatan Ijazah/Transkrip

Ijazah/transkrip yang sudah dibuat akan tersimpan di IPFS dan *hash*-nya akan disimpan pada *Blockchain*. *User* dapat mengunduh *file* ijazah/transkrip melalui menu “Daftar Berkas” seperti pada **Gambar 6**.

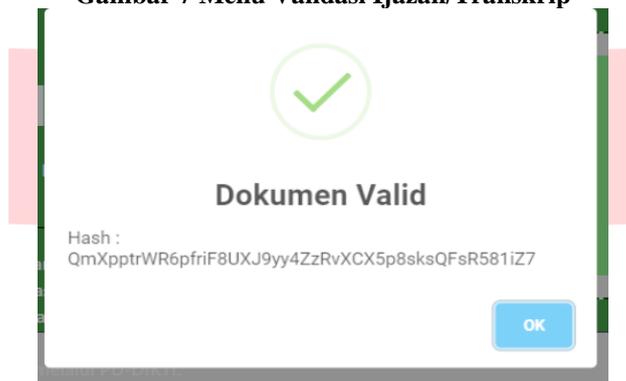


Gambar 6 Menu Daftar Berkas

Menu selanjutnya adalah pengecekan ijazah/transkrip palsu. Pada menu ini, *user* dapat memasukkan dokumen ijazah/transkrip yang ingin dicek keasliannya. Menu pengecekan ijazah/transkrip palsu dapat dilihat pada **Gambar 7**. Jika *file* ijazah/transkrip dinyatakan asli, maka sistem akan menampilkan *hash file* ijazah/transkrip seperti pada **Gambar 8**.



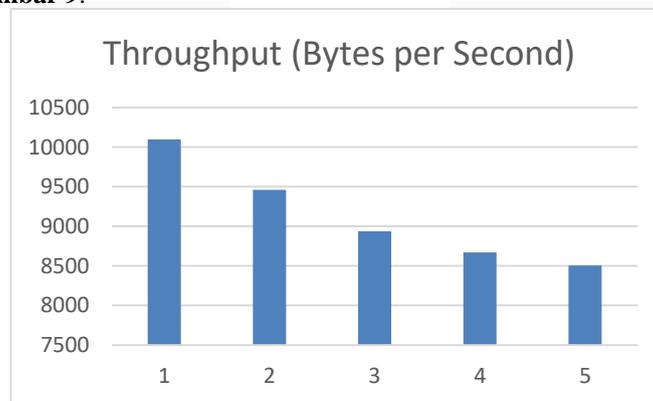
Gambar 7 Menu Validasi Ijazah/Transkrip



Gambar 8 Notifikasi Ijazah/Transkrip Asli

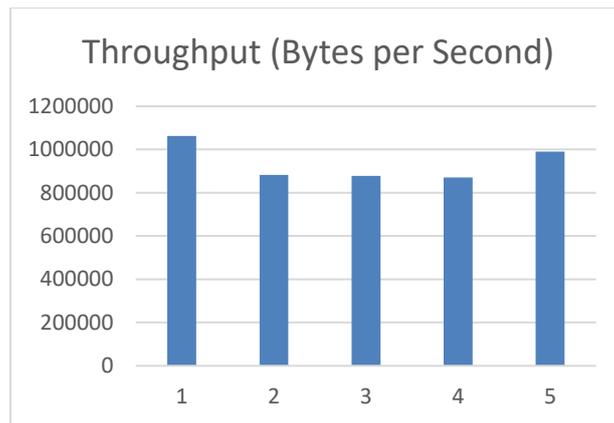
4.2. Analisis QoS dengan Parameter *Throughput*

Throughput merupakan *bandwidth* aktual yang diperoleh ketika melakukan proses pengiriman data. Berdasarkan pengujian yang sudah dilaksanakan, maka diperoleh *throughput* pada saat proses pembuatan ijazah/transkrip pada Gambar 9.



Gambar 9 *Throughput* Proses Pembuatan Ijazah/Transkrip

Berdasarkan Gambar 9, maka dapat dilihat bahwa *throughput* pada pengujian 1 *node* 10.098,5 Bps. *Throughput* pada pengujian 2 *node* 9.462,3 Bps. *Throughput* pada pengujian 3 *node* 8.937,1. *Throughput* pada pengujian 4 *node* 8670,6 Bps dan *Throughput* pada pengujian 5 *node* 8.507 Bps. Selain melakukan pengujian QoS pada proses pembuatan ijazah, dilakukan juga pengujian QoS dengan menggunakan parameter *throughput* pada proses validasi ijazah/transkrip. Hasil QoS pada saat validasi ijazah dapat dilihat pada Gambar 10.



Gambar 10 Throughput Proses Validasi Ijazah/Transkrip

Berdasarkan **Gambar 10**, maka dapat diketahui hasil QoS pada pengujian validasi ijazah/transkrip dengan menggunakan 1 *node* adalah 1.063.100 Bps. *Throughput* pengujian QoS dengan menggunakan 2 *node* adalah 882.700 Bps. *Throughput* pengujian QoS dengan menggunakan 3 *node* adalah 878.500 Bps. *Throughput* dengan menggunakan 4 *node* 871.100 Bps dan *throughput* dengan menggunakan 5 *node* adalah 989.700 Bps.

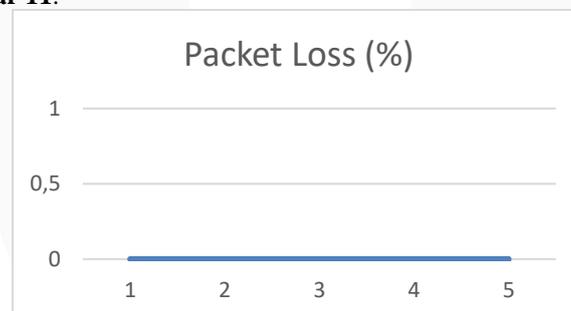
4.3. Analisis QoS dengan Parameter *Packet Loss*

Packet loss merupakan paket yang hilang pada saat proses transmisi ke tujuannya. Menurut *Telecommunications and Internet Protocol Harmonization Over Network (TIPHON)*, *packet loss* dapat dikategorikan seperti pada **Tabel 3**.

Tabel 3 Kategori *Packet Loss*

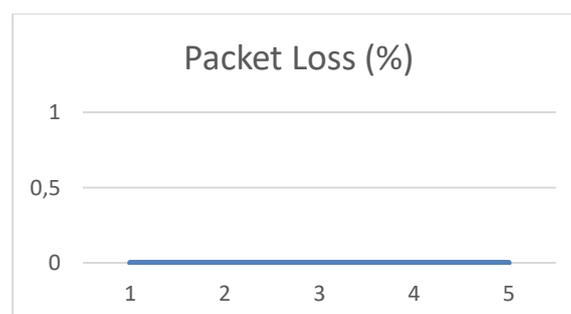
Kategori	<i>Packet Loss</i>
Sangat Bagus	0% - 2%
Bagus	3% - 14%
Sedang	15% - 24%
Buruk	> 25%

Berdasarkan pengujian yang sudah dilaksanakan, maka diperoleh *packet loss* pada saat proses pembuatan ijazah/transkrip pada **Gambar 11**.



Gambar 11 *Packet Loss* Proses Pembuatan Ijazah/Transkrip

Berdasarkan **Gambar 11**, maka dapat dilihat bahwa *packet loss* pada pengujian dengan menggunakan 1 *node*, 2 *node*, 3 *node*, 4 *node* dan 5 *node* bernilai 0%. Oleh karena itu, *packet loss* termasuk kategori sangat bagus. Selain melakukan pengujian QoS pada proses pembuatan ijazah, dilakukan juga pengujian QoS pada proses validasi ijazah/transkrip. Hasil QoS dengan menggunakan parameter *packet loss* pada saat validasi ijazah dapat dilihat pada **Gambar 12**.



Gambar 12 *Packet Loss* Proses Validasi Ijazah/Transkrip

Berdasarkan **Gambar 12**, maka dapat disimpulkan bahwa parameter *packet loss* pada pengujian dengan menggunakan 1 *node*, 2 *node*, 3 *node*, 4 *node* dan 5 *node* bernilai 0% dengan kategori sangat bagus.

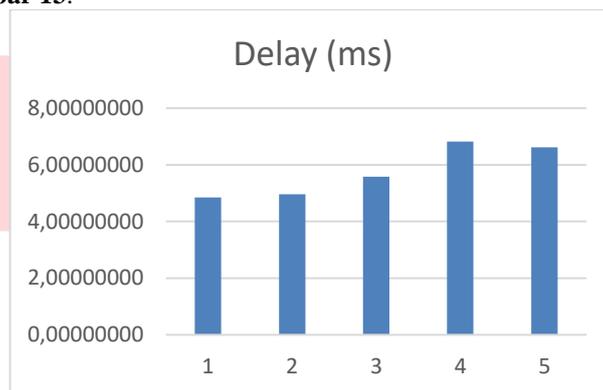
4.4. Analisis QoS dengan Parameter Delay

Delay merupakan waktu yang dibutuhkan untuk mengirimkan data dari *transmitter* ke *receiver*. Menurut *Telecommunications and Internet Protocol Harmonization Over Network (TIPHON)*, *delay* dapat dikategorikan seperti pada **Tabel 4**.

Tabel 4 Kategori Delay

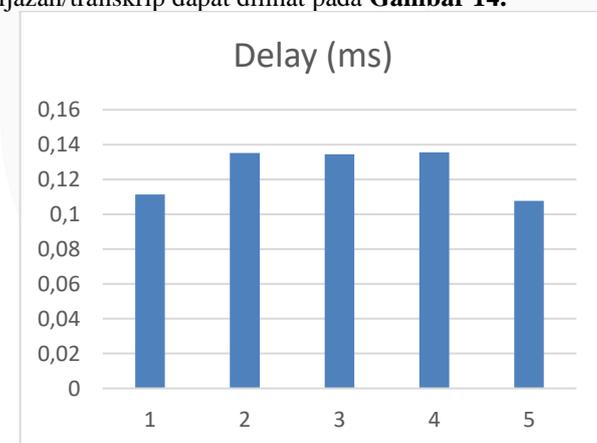
Kategori	Delay
Sangat Bagus	<150 ms
Bagus	150 ms - 299 ms
Sedang	300 ms - 450 ms
Buruk	> 450 ms

Berdasarkan pengujian yang sudah dilaksanakan, maka diperoleh *delay* pada saat proses pembuatan ijazah/transkrip pada **Gambar 13**.



Gambar 13 Delay Proses Pembuatan Ijazah/Transkrip

Berdasarkan **Gambar 13**, maka dapat dilihat bahwa *delay* pada pengujian 1 *node* 4,84 ms. *Delay* pada pengujian 2 *node* 4,96 ms. *Packet loss* pada pengujian 3 *node* 5,57 ms. *Delay* pada pengujian 4 *node* 6,81 ms dan *delay* pada pengujian 5 *node* 6,61 ms. Nilai *delay* pada setiap jumlah *node* termasuk kategori sangat bagus. *Delay* pada proses validasi ijazah/transkrip dapat dilihat pada **Gambar 14**.

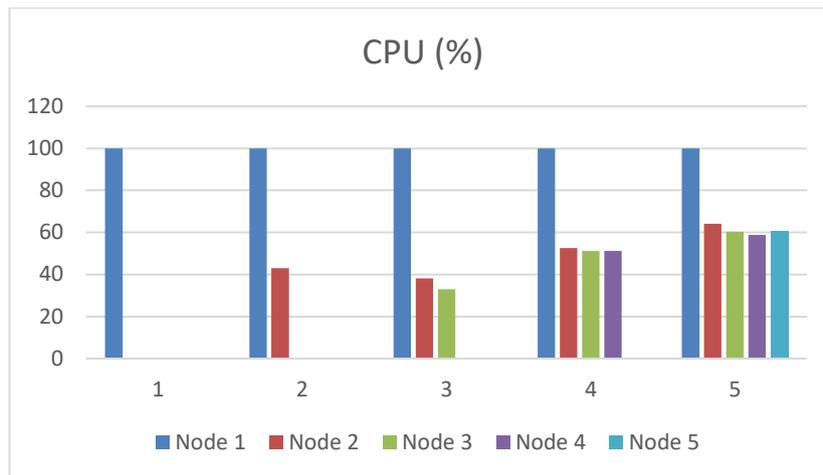


Gambar 14 Delay Proses Validasi Ijazah/Transkrip

Berdasarkan **Gambar 14**, maka dapat dilihat bahwa *delay* pada pengujian 1 *node* 0,111 ms. *Delay* pada pengujian 2 *node* 0,135 ms. *Packet loss* pada pengujian 3 *node* 0,134 ms. *Delay* pada pengujian 4 *node* 0,135 ms dan *delay* pada pengujian 5 *node* 0,107 ms. Nilai *delay* pada setiap jumlah *node* termasuk kategori sangat bagus.

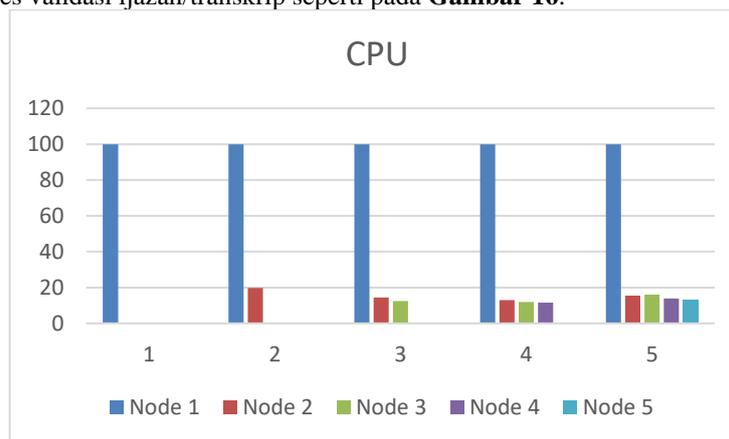
4.5. Analisis Penggunaan CPU

Analisis penggunaan CPU bertujuan untuk mengetahui pengalokasian CPU untuk melakukan proses pembuatan ijazah/transkrip dan proses validasi ijazah/transkrip. Hasil analisis penggunaan CPU dalam proses pembuatan ijazah/transkrip dapat dilihat pada **Gambar 15**.



Gambar 15 Penggunaan CPU Proses Pembuatan Ijazah/Transkrip

Pada **Gambar 15** menunjukkan grafik penggunaan CPU proses pembuatan ijazah/transkrip dalam satuan persen (%). Pada saat pengujian dengan 1 *node*, penggunaan CPU mencapai 100%. Pengujian menggunakan 2 *node*, CPU pada *node* pertama mencapai 100% dan CPU pada *node* kedua mencapai 43,07%. Pengujian menggunakan 3 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua mencapai 38,21% dan *node* ketiga mencapai 32,92%. Pada pengujian dengan 4 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua mencapai 52,54%, *node* ketiga mencapai 51,2% dan *node* keempat mencapai 51,2%. Pengujian dengan menggunakan 5 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua 64,15%, *node* ketiga 60,26%, *node* keempat 58,84% dan *node* kelima 60,41%. Analisis penggunaan CPU juga dilakukan ketika proses validasi ijazah/transkrip seperti pada **Gambar 16**.

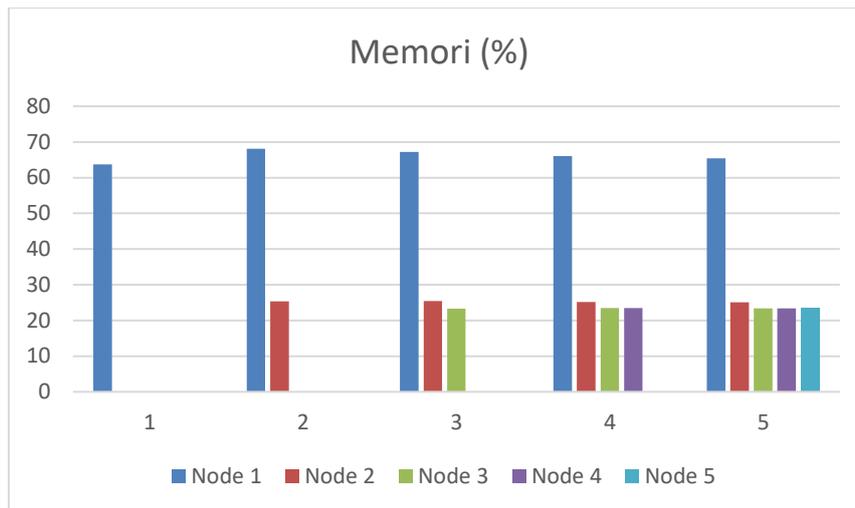


Gambar 16 Penggunaan CPU Proses Validasi Ijazah/Transkrip

Pada **Gambar 16** menunjukkan grafik penggunaan CPU proses validasi ijazah/transkrip dalam satuan persen (%). Pada saat pengujian dengan 1 *node*, penggunaan CPU mencapai 100%. Pengujian menggunakan 2 *node*, CPU pada *node* pertama mencapai 100% dan CPU pada *node* kedua mencapai 19,76%. Pengujian menggunakan 3 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua mencapai 14,37% dan *node* ketiga mencapai 12,39%. Pada pengujian dengan 4 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua mencapai 12,98%, *node* ketiga mencapai 11,93% dan *node* keempat mencapai 11,61%. Pengujian dengan menggunakan 5 *node*, penggunaan CPU pada *node* pertama mencapai 100%, *node* kedua 15,42%, *node* ketiga 16,06%, *node* keempat 13,86% dan *node* kelima 13,28%.

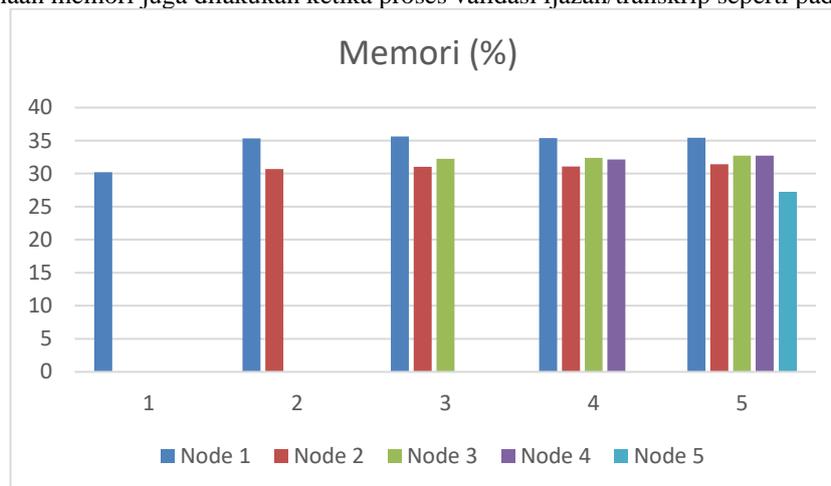
4.6. Analisis Penggunaan Memori

Analisis penggunaan memori bertujuan untuk mengetahui pengalokasian memori untuk melakukan proses pembuatan ijazah/transkrip dan proses validasi ijazah/transkrip. Hasil analisis penggunaan memori dalam proses pembuatan ijazah/transkrip dapat dilihat pada **Gambar 17**.



Gambar 17 Penggunaan Memori Proses Pembuatan Ijazah/Transkrip

Pada **Gambar 17** menunjukkan grafik penggunaan memori proses pembuatan ijazah/transkrip dalam satuan persen (%). Pada saat pengujian dengan 1 *node*, penggunaan memori mencapai 63,7%. Pengujian menggunakan 2 *node*, memori pada *node* pertama mencapai 68,09% dan memori pada *node* kedua mencapai 25,32%. Pengujian menggunakan 3 *node*, penggunaan memori pada *node* pertama mencapai 67,16%, *node* kedua mencapai 25,41% dan *node* ketiga mencapai 23,3%. Pada pengujian dengan 4 *node*, penggunaan memori pada *node* pertama mencapai 66,07%, *node* kedua mencapai 25,2%, *node* ketiga mencapai 23,44% dan *node* keempat mencapai 23,44%. Pengujian dengan menggunakan 5 *node*, penggunaan memori pada *node* pertama mencapai 65,4%, *node* kedua 25,07%, *node* ketiga 23,42%, *node* keempat 23,4% dan *node* kelima 23,67%. Analisis penggunaan memori juga dilakukan ketika proses validasi ijazah/transkrip seperti pada **Gambar 18**.



Gambar 18 Penggunaan Memori Proses Validasi Ijazah/Transkrip

Pada **Gambar 18** menunjukkan grafik penggunaan memori proses validasi ijazah/transkrip dalam satuan persen (%). Pada saat pengujian dengan 1 *node*, penggunaan memori mencapai 30,2%. Pengujian menggunakan 2 *node*, memori pada *node* pertama mencapai 35,34% dan memori pada *node* kedua mencapai 30,7%. Pengujian menggunakan 3 *node*, penggunaan memori pada *node* pertama mencapai 35,61%, *node* kedua mencapai 31,02% dan *node* ketiga mencapai 32,26%. Pada pengujian dengan 4 *node*, penggunaan memori pada *node* pertama mencapai 35,39%, *node* kedua mencapai 31,08%, *node* ketiga mencapai 32,4% dan *node* keempat mencapai 32,13%. Pengujian dengan menggunakan 5 *node*, penggunaan memori pada *node* pertama mencapai 35,42%, *node* kedua 31,41%, *node* ketiga 32,72%, *node* keempat 32,75% dan *node* kelima 27,27%.

5. Kesimpulan

Sistem pendeteksi ijazah dan transkrip merupakan sistem yang berfungsi untuk membuat ijazah dan transkrip serta menguji keasliannya dengan menggunakan *smart contract Blockchain* dan *Interplanetary File System (IPFS)*. IPFS digunakan untuk menyimpan *file* ijazah/transkrip secara terdistribusi. Dengan menggunakan metode terdistribusi ini dapat mencegah penyerangan SQLInjection yang rentan terjadi pada penyimpanan terpusat. Pada penelitian ini juga dilakukan analisis *Quality of Service (QoS)* pada saat pembuatan ijazah/transkrip dan proses validasi ijazah/transkrip. Parameter QoS yang digunakan pada pengujian ini adalah *throughput*, *packet loss* dan *delay*. Berdasarkan percobaan yang sudah dilakukan, nilai *throughput* terbaik pada

saat proses pembuatan ijazah/transkrip adalah dengan menggunakan 1 *node* yaitu 10.098,5 Bps. Nilai *throughput* terbaik pada saat validasi ijazah/transkrip adalah dengan menggunakan 1 *node* yaitu 1.063.100 Bps. Nilai *packet loss* pada proses pembuatan ijazah/transkrip dan validasi ijazah/transkrip termasuk kategori sangat baik. Nilai *delay* pada proses pembuatan ijazah/transkrip dan validasi ijazah/transkrip memiliki kategori sangat baik. Berdasarkan analisis penggunaan memori dan CPU, sistem ini mampu dijalankan baik menggunakan 1 *node* sampai dengan 5 *node*.

Penelitian yang dilakukan menggunakan perangkat virtual. Oleh karena itu, untuk pengembangan penelitian selanjutnya dapat menggunakan perangkat fisik sehingga penggunaan CPU dan memori dapat dianalisis dengan lebih akurat. CPU yang disarankan untuk penelitian selanjutnya juga memiliki spesifikasi lebih tinggi dibandingkan dengan penelitian ini.

REFERENSI

- [1] KBBI, "KBBI Daring," Kemdikbud, 2016. [Online]. Available: kbbi.kemdikbud.go.id. [Accessed 2020 Oktober 2020].
- [2] I. Teguh, "Mengapa Orang Indonesia Memalsukan Ijazah?," *tirto.id*, 27 November 2018. [Online]. Available: <https://tirto.id/mengapa-orang-indonesia-memalsukan-ijazah-dau1>. [Accessed 4 Oktober 2020].
- [3] Kemenristekbrin, "Cegah dan Antisipasi Ijazah Palsu dengan SIVIL PIN," Kementerian Riset dan Teknologi/Badan Riset dan Inovasi Nasional, 2 Mei 2016. [Online]. Available: <https://www.ristekbrin.go.id/kabar/cegah-dan-antisipasi-ijazah-palsu/>. [Accessed 4 Oktober 2020].
- [4] B. K. Mohanta, S. S. Panda and D. Jena, "An Overview of Smart Contract and Use cases in Blockchain Technology," in *9th ICCCNT 2018 IEEE*, India, 2018.
- [5] A. M. Aziz, A. Budiono and A. Widjajarto, "Analisis dan Implementasi Komunikasi Antar Node IPFS (Interplanetary File System) pada Smart Contract Ethereum," *e-Proceeding of Engineering*, vol. 6, no. 2, pp. 7670 - 7678, 2019.
- [6] M. Steichen, B. Fiz, R. Norvill, W. Shbair and R. State, "Blockchain-Based, Decentralized Access Control for IPFS," *IEEE*, pp. 1499-1506, 2018.
- [7] A. H. Nugroho, "Validasi Ijazah dengan Menggunakan Watermarking," pp. 9 - 15.
- [8] F. Nuraeni, Y. H. Agustin and I. M. Muharam, "Implementasi Tanda Tangan Digital Menggunakan RSA dan Implementasi Tanda Tangan Digital Menggunakan RSA dan," in *Konferensi Nasional Sistem Informasi*, Pangkalpinang, 2018.
- [9] Mendikbud, "Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 8 1 Tahun 2014," LLDIKTI, 2014.
- [10] D. Rustam, "Penomoran Ijazah Nasional & Sistem Verifikasi Ijazah Secara Elektronik," Kemenridtekdikti, Malang, 2018.
- [11] Y. Chen, H. Li, K. Li and J. Zhang, "An improved P2P File System Scheme based on IPFS and Blockchain," *IEEE*, pp. 2652 - 2657, 2017.
- [12] N. Nizamuddin, K. Salah, M. A. Azad, J. Arshad and M. Rehman, "IPFS, Decentralized document version control using ethereum blockchain and," *Computers and Electrical Engineering*, pp. 183-197, 2019.
- [13] A.-T. Pănescu and V. Manta, "Smart Contracts for Research Data Rights Management over the Ethereum Blockchain Network," Taylor & Francis Group, Romania, 2018.
- [14] G. Wood, "Ethereum: A Secure Decentralised Generalised Transaction Ledger," 2017. [Online]. Available: <https://gavwood.com/paper.pdf>. [Accessed 16 10 2020].
- [15] G. G. Dagher, P. B. Marella, M. Milojkovic and J. Mohler, "BroncoVote: Secure Voting System using Ethereum's Blockchain," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Colorado, 2018.
- [16] D. Macrinici, "Smart Contract Applications within Blockchain Technology: A Systematic Mapping Study," *Telematics and Informatics*, 2018. [Online]. Available: doi: <https://doi.org/10.1016/j.tele.2018.10.004>.
- [17] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System," 14 7 2014. [Online]. Available: <https://arxiv.org/pdf/1407.3561.pdf>. [Accessed 16 10 2020].
- [18] S. Muralidharan and H. Ko, "An InterPlanetary File System (IPFS) based IoT framework," *IEEE International Conference on Consumer Electronics (ICCE)*, 2019.
- [19] R. Dwinanda, "Cegah Pemalsuan, PTS Diajak Ajukan Penomoran Ijazah Nasional," *Republika*, 26 Agustus

2019. [Online]. Available: <https://republika.co.id/berita/pwudct414/cegah-pemalsuan-pts-diajak-ajukan-penomor-an-ijazah-nasional>. [Accessed 4 Oktober 2020].
- [20] Blockchain.com, "Jumlah total transaksi pada blockchain," [Online]. Available: <https://www.blockchain.com/charts>. [Accessed 16 10 2020].
- [21] T. Novianti and A. Widiatoro, "Analisa QOS (Quality of Services) pada Implementasi IPV4 dan IPV6 dengan Teknik Tunneling," *Jurnal Ilmiah Rekayasa*, vol. 9, no. 2, pp. 76-83, 2016.

