

## **Perancangan Pengamanan Situng Form C1 Berbasis Input Desa Dan Output Kecamatan Menggunakan Blockchain**

**Warna Agung Cahyono<sup>1</sup>, Sri Anggraeni Kusuma Dewi<sup>2</sup>**

<sup>1</sup>Institusi Teknologi Dan Bisnis Asia

<sup>2</sup> Institusi Teknologi Dan Bisnis Asia

email : <sup>1</sup>warnaagung@asia.ac.id, <sup>2</sup>kusumadewi@asia.ac.id

**ABSTRAK.** Kasus pemilihan presiden tahun 2019 menyisakan duka meninggalnya beberapa KPPS karena faktor kelelahan. Ada dua kegiatan yang harus dilakukan KPPS yaitu penghitungan di TPS dan antrian di kelurahan/kecamatan. Setelah penghitungan mereka melakukan antri penyerahan dokumen di kelurahan atau kecamatan. Untuk memangkas waktu kegiatan, peneliti mengusulkan penggunaan aplikasi mobile untuk mengurangi antrian di kelurahan/kecamatan. Tetapi hal tersebut juga memunculkan masalah tambahan, yaitu faktor keamanan atau kepercayaan pengguna mobile setelah data tiba di database kelurahan. Pada masalah tambahan ini peneliti menggunakan konsep blockchain. Blockchain menggunakan kunci hash untuk mengenkripsi satu blok data. Satu blok data berisi satu data penuh form C1. Satu blok data ditambah nonce dijadikan satu dokumen kemudian dienkripsi menggunakan SHA-512 untuk menghasilkan hash. hash tersebut ditambah blok berikutnya dan nonce berikutnya dijadikan satu dokumen untuk dienkripsi sehingga menghasilkan hash berikutnya lagi. Proof of work di dalam blockchain menggunakan validasi pada blok rantai yang panjang. Nonce ini berfungsi untuk memperlambat proses chaining ulang dengan cara membangkitkan jumlah digit 0 awal hash. Sehingga dengan demikian data tidak dapat dilakukan perubahan ulang dalam waktu cepat. Pengujian dilakukan dengan dua cara yaitu input dari TPS melalui server desa hingga menampilkan data di sisi server kecamatan. Kemudian uji chaining pada 1 digit 0 awal hash.

**Kata Kunci:** *SITUNG; Form C1; Blockchain*

### **ABSTRACT.**

The case of the 2019 presidential election leaves the grief of the death of several KPPS due to fatigue factor. There are two activities that kpps must do, namely counting at tps and queues in the kelurahan / kecamatan. After counting they queue for submission of documents in the kelurahan or kecamatan. To cut off the time of activities, researchers propose the use of mobile applications to reduce queues in the kelurahan / kecamatan. But it also raises additional problems, namely the security factor or trust of mobile users after the data arrives in the kelurahan database. On this additional issue researchers use the concept of blockchain. The blockchain uses hash keys to encrypt a single block of data. One block of data contains one full form data C1. One block of data and nonce are added into one document. Then it is encrypted using SHA-512 to generate a hash. the hash, the next block and the next nonce are added become one document to encrypt, resulting in the next hash again. Proof of work within the blockchain uses validation on long chain blocks. This nonce serves to slow down the re-chaining process by generating a number of 0's digit as the prefix hash. Thus the data can not be changed again in a fast time. Testing is done in two ways, namely input from TPS through kelurahan server to display data on the server side of the kecamatan. Then test chaining at the 1 digit of 0 prefix hash

**Keywords:** *SITUNG; Form C1; Blockchain*

### **PENDAHULUAN**

Hajat besar negara Indonesia pada Pemilu 2019 telah usai. Namun menyisakan duka mendalam pada beberapa petugas KPPS (Alam, 2019). Petugas KPPS adalah petugas yang memungut dan menghitung suara di TPS. TPS adalah tempat pelaksanaan pemungutan dan penghitungan suara (KPU, 2019). Di dalam buku panduan tersebut petugas KPPS merekapitulasi hasil penghitungan suara saat itu juga. Selanjutnya hasil suara penghitungan dikumpulkan di PPK kecamatan. Petugas PPS bekerja ekstra pada dua kejadian ini. Yaitu kejadian penghitungan suara di TPS dan kejadian antrian panjang di kelurahan/desa.

Pada penelitian kali ini, penulis memilih mengangkat permasalahan pada antrian yang seharusnya tidak perlu terjadi di PPK. Permasalahan kedua yang diangkat adalah pengamanan data di tingkat PPK. Pengamanan data C1 dilakukan untuk menghindari kemungkinan perubahan data di PPK dapat terlacak.

Pada Gambar 1 antrian terjadi di PPK kecamatan disebabkan oleh berkumpulnya semua TPS di kecamatan tersebut (KPU, 2019). Setiap dusun dapat memiliki lebih dari satu TPS mengikuti rasio jumlah pemilih per TPS (Bawaslu, 2019). Sehingga TPS harus menyampaikan semua formulir C, C1-KPU, C1-PPWP, C1-DPR, C1-DPD, C1-DPRD Propinsi, dan C1-DPRD Kab/Kota kepada PPK di kecamatan. Tapi pada

penelitian ini, penulis berfokus pada transaksi pengiriman dan penyimpanan data form C1-PPWP. Mengikuti kejadian pada pemilu 2019, salah satu masalah petugas KPPS berupa antrian dalam hal memberikan suara dan menyetorkan hasil pemungutan suara hendaknya tidak terjadi lagi.



**Gambar 1.** Alur Hitung Dan rekap suara

Sesuai dengan pasal 61 ayat 1 PKPU No.3 Tahun 2019, maka KPPS dapat mengumumkan salinan C1 yang dapat diakses publik selama 3 hari. Sehingga memungkinkan terjadinya transparansi publik. Beberapa sumber berita memuat laporan oleh Bawaslu terkait adanya perubahan data tingkat PPS, salah satunya mengenai perbedaan suara versi KPU dan Bawaslu (Farisa, 2019). Gugatan ini akan menyebabkan pekerjaan tambahan sehingga menambah faktor kelelahan para petugas. Berita kematian para petugas karena faktor kelelahan seharusnya dapat dihindari jika data adalah valid tanpa perubahan.

Sehingga dua permasalahan tersebut diangkat oleh penulis. Masalah mengenai antrian petugas TPS dalam menyerahkan salinan form C1. Masalah berikutnya adalah keamanan data yang disetorkan di PPK. Untuk menangani masalah antrian penyerahan form C1 para petugas tps ke PPS. Dan untuk menangani masalah perubahan data setelah diserahkan ke PPS, maka peneliti mengusulkan untuk pengiriman data form C1 dengan mobile android sekaligus mengajukan solusi pengamanan data jika dirubah bahkan oleh admin database sekalipun. Peneliti mengusulkan rancangan system mobile android C1 dapat mengirim pada data form C1 service API server desa sekaligus mengunci datanya setelah data masuk di server.

Solusi menggunakan aplikasi berbasis desktop dengan jaringan, web, atau android maupun desktop memiliki kelebihan dan kerangan masing-masing. Implementasi dengan metode desktop berbasis jaringan harus install terlebih dahulu dan hanya dapat diakses melalui komputer. Implementasi berbasis web dapat diakses melalui komputer desktop maupun smartphone. Sedangkan implementasi berbasis android dapat diakses hanya melalui handphone. Dari sudut pandang jumlah peminatan, menurut Paul Fontelo aplikasi android lebih diminati daripada aplikasi web (Fontelo & Liu, 2013). Terbukti bahwa penggunaan aplikasi mobile android dapat meningkatkan jumlah aktifitas perilaku pencarian dan penggunaan tool-tool yang disediakan oleh programmernya (Fontelo & Liu, 2013). Dengan demikian edukasi dan pelatihan kepada calon pengguna aplikasi berbasis android lebih mudah dari pada melatih penggunaan web kepada pengguna. Disamping itu aplikasi mobile berbasis android memiliki kecepatan lebih daripada mobile web (Yoon, 2012).

Sehingga dengan demikian solusi yang akan digunakan antara penggunaan web, desktop, dan mobile android, maka penulis memilih menggunakan mobile android. Masalahnya adalah antrian yang terjadi di setiap PPS di setiap kelurahan/desa dalam kecamatan. Mari kita melihat contoh pada kecamatan Sukun kota Malang. Menurut data dari website <https://pemilu2019.kpu.go.id/#/ppwp/hitung-suara/>, jumlah TPS di kecamatan Sukun terdapat 5 kelurahan. Salah satu kelurahan seperti Kelurahan Bandulan terdapat 45 TPS. Pada pilpres 2019 setiap TPS terdiri dari 5 orang terdiri dari 2 saksi kedua paslon, 1 KPPS, 1 Ketua TPS, dan 1 perwakilan anggota KPU (KPU B. T., 2019). Sehingga total dalam 1 PPS kelurahan Bandulan terdapat 225 pengguna.

Di PPK Sukun terdapat 11 PPS dengan 453 TPS. Karena pada pilpres 2019 terdapat dua paslon sehingga dalam 1 tps terdapat 5 orang sehingga total terdapat 2265 pengguna di PPK Sukun. Semua PPK ini dapat mengirim bersamaan ke server. Pengiriman secara bersamaan ini merupakan masalah berikutnya dikarenakan memungkinkan terjadi *concurrency deadlock*. Sehingga basis pengiriman pengguna tertuju pada

aplikasi per PPS desa/kelurahan. Jumlah TPS terbanyak di PPK Sukun ada di PPS Bandungrejosari sejumlah 89 TPS.

Para pengguna adalah 5 orang dari kelompok perwakilan setiap TPS. Sehingga pada sisi client ada 5 pengguna aplikasi per TPS di dalam satu kelurahan/desa. Jumlah Jika ditotal akan ada 5 client dikali jumlah TPS 89 maka terdapat 445 aplikasi client yang terhubung ke server PPS kelurahan Bandungsari.

Masalah kedua setelah masalah pengiriman data C1 adalah pengamanan data di server PPS. Mengamankan data di database supaya tidak dapat dirubah oleh siapapun bukan perkara mudah. Karena ada beberapa potensi perubahan yaitu penyusupan hacker luar, hacker dari dalam, programmer aplikasi, maupun administrator databasenya sendiri. Blockchain (Nakamoto, 2009) menawarkan suatu metode untuk menghambat bahkan membuat sesuatu menjadi tidak dapat dirubah secara matematis. Pada setiap rantai blok terdapat nonce, semakin panjang angka 0 di depannya maka akan semakin lama pula proses untuk modifikasi seluruh rantai data.

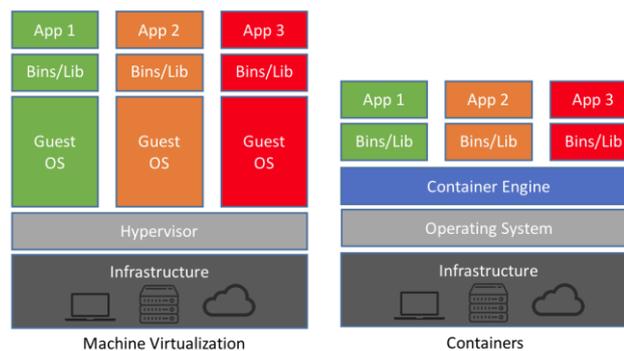
## 2. METODE PENELITIAN

### 2.1 Network Latency Pada Web Browsing

*Latency* adalah jeda waktu keterlambatan antara permintaan pengguna dengan response http tujuannya. Request http yang akan digunakan adalah POST dan GET. Pada sisi server terdapat beberapa tahapan sebelum client mendapatkan *response http*. Tahapan yang diproses di server adalah pertama bagian webserver( seperti apache, nginx, JSP, dsb), kedua bahasa pemrogramna(PHP, python, golang, java, dsb), dan ketiga database yang digunakan. Dari sisi web server, apache web server memiliki latency yang meningkat seiring dengan peningkatan jumlah koneksi perdetik (Satwika & Semadi, 2020). Pada tahap proses bahasa pemrograman sisi server maka dapat dihitung dengan pendekatan kompleksitas waktu untuk semua API yang dibutuhkan. Kompleksitas waktu yang peneliti gunakan adalah  $O(1)$ , dapat dilihat pada <https://github.com/warnaagung/dbkpu.git>. Tahap berikutnya adalah waktu reponse terkait operasi database mariaDB. Sebagaimana pada operasi web server apache, maka mariaDB juga memiliki peningkatan latency seiring dengan jumlah permintaan koneksi perdetik (Warman & Ramdaniansyah, 2018). Waktu *response* pada mariaDB meningkat signifikan dengan jumlah operasi yang terjadi.

### 2.2 Container(Platform Kubernetes)

Container adalah bagian software yang dapat dieksekusi yang mengemas kode aplikasi beserta dependensinya, dan memungkinkan berjalan di atas berbagai konfigurasi sistem operasi yang ditumpanginya. Bisa dikatakan wadah independen yang terpisah dari sistem operasi host (OS).



**Gambar 2.** Perbandingan Virtual Machine Dengan Container

Sesuai gambar 2. jika dibandingkan dengan VM(*Virtual Machine*) yang memuat *guest OS(Operating System)*, maka *container* memuat *library* dan dependensinya. VM memanfaatkan *hypervisor*(suatu layer antara VM dengan *hardware*) untuk memvirtualisasikan hardware fisik tapi berisi duplikasi penuh dari sistem operasi. Dengan memvirtualisasikan hardware maka sistem operasi di dalam VM dapat menjalankan aplikasi yang independen dari host OS.

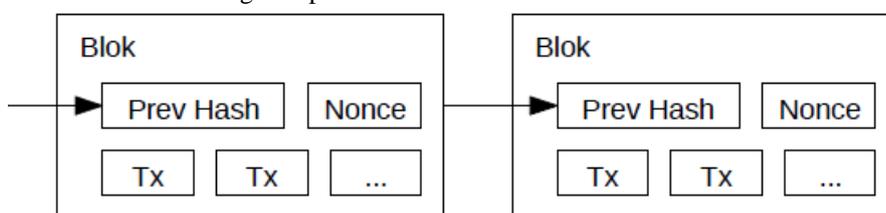
Hal yang sama juga dilakukan oleh container, tetapi container tidak berasal dari duplikasi sistem operasi untuk memvirtualisasikan hardware dan sistem operasi. Container hanya berisi aplikasi, *library*, dan dependensinya. Sehingga dari sisi ukuran penggunaan *disk storage*, *CPU*, dan *memory(RAM)* lebih kecil daripada VM.

Salah satu container yang digunakan peneliti adalah *kubernetes*. *Kubernetes* adalah bersifat open source oleh google. *Kubernetes* berfungsi sebagai platform untuk otomasi *deployment*, menejemen, dan *scaling*. Sehingga kelak satu host OS dapat menampung kecamatan yang didalamnya berisi container untuk setiap desa/kelurahan.

**2.2 Blockchain**

Blockchain adalah kumpulan blok-blok rantai yang dapat disimpan di database meski server database tersebut bersifat publik. Rahasia pengamanannya ada pada rantai hash. Satu blok dari rantainya dilakukan fungsi hash. Hasil dari penghitungan hash dimasukkan kedalam blok berikutnya. Jadi setiap blok menyertakan penghitungan hash blok sebelumnya, kecuali blok pertama.

Untuk mempersulit atau memperlambat proses pembuatan sebuah blok, maka nilai hash penghitungan dikodekan memiliki beberapa digit angka nol. Pada umumnya hasil penghitungan hash berupa angka acak, dan tidak selalu mengandung angka nol pada digit pertama. Untuk membentuk angka nol pada digit pertama, diperlukan nonce. Mining(penggalian) angka nol pada satu buah blok ini bisa memakan waktu yang lama. Dengan demikian diharapkan jika suatu rantai blockchain terbentuk panjang, maka tidak ada kesempatan seseorang merubah rantai tersebut dengan cepat.



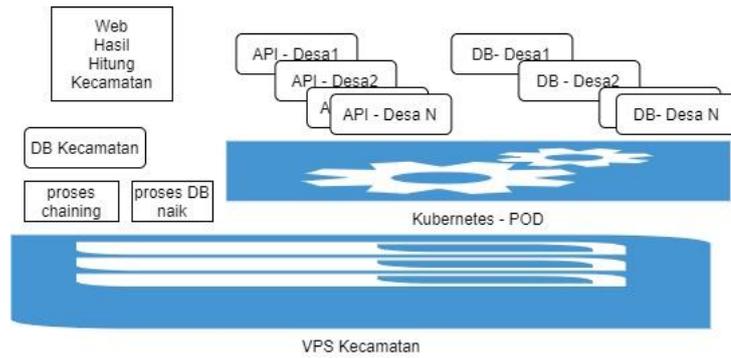
**Gambar 3.** Rantai Dengan Dua Blok

*Nonce* adalah variabel bertipe numerik. *Nonce* setiap blok disesuaikan dengan jumlah angka 0 di depan *nexthash*. Pada Gambar 3, terdapat contoh 2 blok dari rantai blockchain. Pada blok sebelah kiri jika dilakukan fungsi *hash* maka hasilnya adalah *nexthash*. *Nexthash* hasil penghitungan hash pada blok tersebut menjadi *prevhash* pada blok setelahnya. Sehingga akan selalu membentuk rantai yang terhubung. Semakin panjang angka 0 depan pada setiap hash maka akan semakin sulit pula perubahan pada semua blok.

Mekanisme dengan memanfaatkan panjang digit 0 awal di setiap hash pada semua blok pada rantai disebut sebagai proof of work. Hal ini bertujuan untuk memperlambat perubahan pada suatu blok rantai. Jika petugas TPS menambah data, maka sistem melakukan komputasi pada nonce untuk menemukan hash dengan digit 0 diawal. Lamanya waktu yang dibutuhkan untuk membuat satu buah blok menjadi kelebihan tersendiri pada blockchain. Sehingga jika seseorang berusaha untuk merekayasa data, maka dia harus merubah semua hash pada semua blok sepanjang sisa rantai. Sehingga hal itu tidak mungkin dilakukan dalam waktu pendek. Jika satu blok memerlukan 10 menit, maka jika ada 1000 blok maka diperlukan 10000 menit atau 6 hari 22 jam.

**2.3 Diagram Infrastruktur**

Konsep yang digunakan di sini adalah aplikasi beserta database tersebar dan visualisasi data terpusat. Konsep ini dilakukan sebagai cara yang aman untuk menghindari adanya kemungkinan latency interaksi ketika proses input oleh para petugas TPS secara serempak di semua daerah di Indonesia. Visualisasi data pusat yang bersifat hanya readonly. Data pusat adalah pusat informasi yang berfungsi memonitor hasil penghitungan suara baik oleh KPU maupun oleh masyarakat luas yang ingin melihat secara online. Sehingga konsep penelitian memadukan input dari desa/kelurahan dan monitor/baca hasil dari kecamatan beserta keamanan data dari modifikasi di kemudian hari.

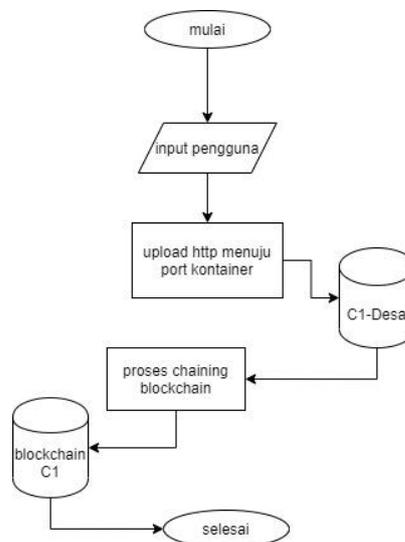


Gambar 4. Diagram Arsitektur

Pusat visualisasi data ada pada aplikasi web server *Web Hasil Hitung Kecamatan* pada Gambar 4. Visualisasi ini menampilkan data seluruh form C1 pada kecamatan tersebut. Data seluruh form C1 tingkat kecamatan ada di database *DB Kecamatan* pada Gambar 4. Pada Gambar 4, *proses DB naik*, adalah aplikasi untuk membaca dan memindah database dari DB desa 1 sampai desa ke N. Aplikasi ini berjalan dibalik layar. Pada Gambar tersebut, DB – desa1 sampai DB-DesaN diisi oleh API-Desa 1 sampai dengan API-Desa N. Setiap port pada masing-masing api di *mapping* ke port sistem operasi host. Sehingga pada arsitektur tersebut tiga aplikasi dan satu database berada di VPS. Kemudian di dalam POD kubernetes terdapat kontainer API server sebanyak sejumlah desa yang ada, dan terdapat kontainer database mariaDB sebanyak jumlah desa yang ada.

**2.4 Alur Sistem**

Konsep ini meringkas beberapa server desa menjadi satu server. Pendekatan yang dilakukan menggunakan kontainer. Sehingga satu server kecamatan di dalamnya terdiri dari beberapa server desa. Setiap server desa melayani komunikasi dari beberapa beberapa pengguna.



Gambar 5. Alur Input Sistem

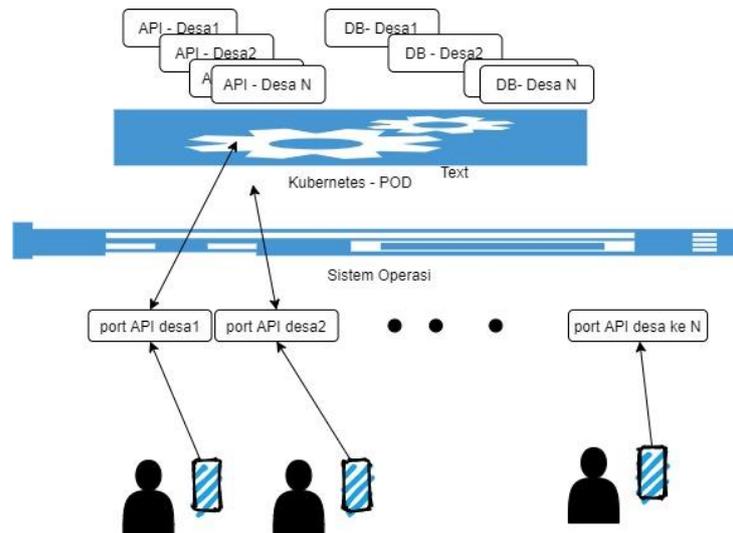
Pada Gambar 5 di atas, pengguna memasukkan data menggunakan mobile android. Kemudian pengguna menyimpan data. Mobile android melakukan request POST koneksi http menuju ke kontainer server desa. Selanjutnya aplikasi di server kecamatan membaca ke kontainer desa. Aplikasi membaca database kemudian melakukan chaining pada setiap data C1 yang masuk menuju ke desa.

**2.5 Sistem Input Desa**

Sistem input desa menggunakan container kubernetes sebagai pengganti server di setiap desa. Satu buah kecamatan diwakili oleh mesin fisik atau mesin virtual dan setiap desa/kelurahan diwakili oleh dua

kontainer yaitu satu image untuk database mariadb dan satu image untuk web server berisi API PHP. Setiap port pada webserver *dimapping* ke port host sistem operasi. Kota Malang terdiri dari 5 kecamatan, sehingga jika diterapkan penuh maka membutuhkan 5 server atau 5 server virtual. Setiap kecamatan terdiri dari beberapa desa/kelurahan. Setiap kelurahan berupa dua buah container yaitu API dalam bahasa PHP dan database MariaDB. Karena container tidak menyalin/menggandakan sistem operasi maka satu buah server kecamatan berupa satu sistem operasi server.

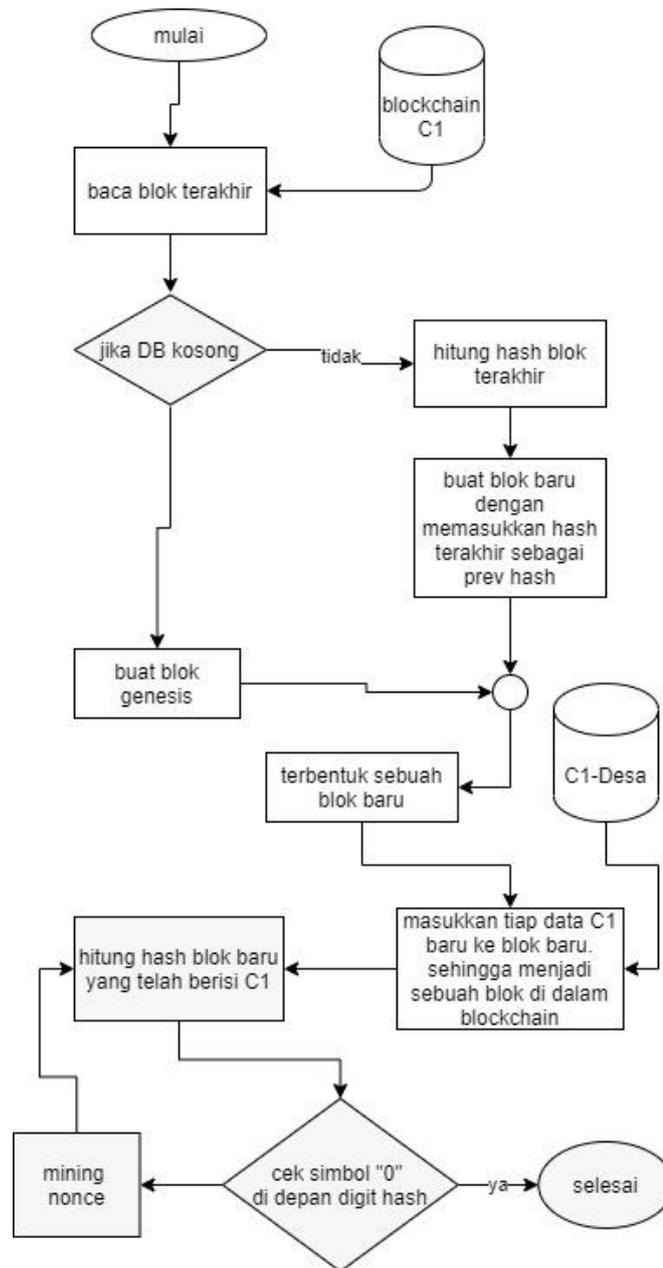
Sebagai contoh kecamatan Blimbing kota Malang memiliki 10 kelurahan. Maka jika setiap kelurahan memerlukan 2 kontainer(database dan web server) maka satu server kecamatan blimbing memiliki 20 image container kubernetes.



**Gambar 6.** Model Input Pengguna ke port kecamatan menuju ke kontainer desa

Pada gambar tersebut, setiap pengguna masuk melalui port host sistem operasi yang dimapping ke port container. Sehingga pengguna dapat berinteraksi dengan web server container. Pada tahap berikutnya web container menyimpan data dari pengguna ke database container yang ditentukan.

Setelah tahap entri data selesai. Sistem pada server kecamatan melakukan chaining data kelurahan chaining dilakukan pertama kali pada satu database kelurahan, hal ini untuk pelaksanaan konsep blockchain *proof of work* pertama kali. Selanjutnya tahap berikutnya chaining diduplikasi juga di kecamatan dan secara paralel dilakukan chaining secara independen dari kelurahan. Sehingga terdapat dua chain utama yaitu tingkat kelurahan dan tingkat kecamatan. Chain terpanjang berada di kecamatan. Rantai blockchain panjang dikelurahan ini yang dapat dilihat oleh kpu maupun oleh masyarakat.



Gambar 7. Proses Blockchain

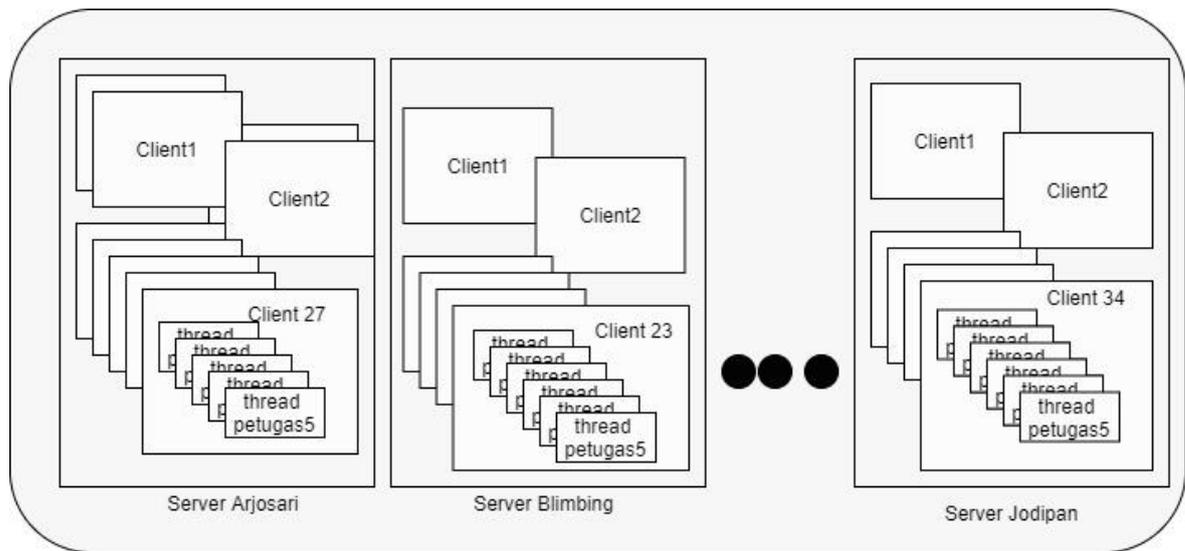
Pada gambar 7 diatas, aplikasi java membaca database dari server setiap kelurahan dan menyusun rantai blockchain pada server kecamatan. Dikarenakan lamanya rantai terbentuk, maka pada tahap awal hanya menggunakan satu digit 0 pada setiap hash dari bloknya. Blok terakhir pada rangkaian semua rantai blockchain database chain kelurahan akan dihitung hashnya. Hash yang dihitung menjadi *previous hash* pada blok *genesis* pada blockchain kecamatan. *Blok genesis* adalah blok pertama dalam rantai blockchain.

Pada Gambar 7 diatas, proses diawal dengan membaca data blockchain, jika kosong maka buat blok genesis. Tapi jika sudah terisi, maka lakukan hash pada blok terakhir. Kemudian buat blok baru dan masukkan hasil penghitungan hash. Baik blok genesis maupun blok baru berisi hash blok sebelumnya sebenarnya adalah sebuah blok baru. Blok baru tersebut berisi data C1. Jika blok baru dihitung hash tidak menandung angka nol di awal digit maka lakukan proses pencarian angka nonce. Perubahan angka nonce berdampak pada perubahan nilai hash. Jika nilai hash mengandung angka nol di depan maka proses berhenti dan blok baru siap ditambahkan ke blockchain

**3. PEMBAHASAN**

**3.1 Uji Input**

Pengujian input dilakukan dengan dua cara. Yaitu menggunakan mobile phone dan menggunakan java post. Skenario yang dilakukan adalah menggunakan cloud. Uji menggunakan *compute engine* dari GCP(Google Cloud Platform). Jumlah *compute engine* yang dibutuhkan sejumlah kelurahan yang ada. Pada saat ini menggunakan kasus kecamatan Blimbing kota Malang. Jumlah kelurahan di kecamatan Blimbing ada 10 kelurahan, sehingga ada 10 compute engine.



**Gambar 8.** skenario Uji menggunakan robot entri menggunakan java http

Peneliti menyewa GCP yang terdiri dari 10 compute engine mulai dari arjosari, blimbing, sampai dengan jodipan. Setiap compute engine menjalankan aplikasi sebanyak sejumlah TPS yang ada. Dan setiap aplikasi menjalan 5 thread koneksi http ke server desa. Setiap thread mewakili satu orang pengguna. Karena di setiap tps terdiri dari 5 petugas, maka jumlah thread untuk pengujian adalah 5.

**Tabel 1.** Uji koneksi http pada tanggal 14 oktober 2020

No	Kelurahan	Jumlah TPS	user	Total <i>concurrent</i> koneksi	Rata-rata Latensi (menit)
1	Kelurahan Baleaarjosari	24	5	120	2,5
2	Kelurahan Arjosari	27	5	135	2,6
3	Kelurahan Polowijen	31	5	155	2,8
4	Kelurahan Purwodadi	49	5	245	4,1
5	Kelurahan Pandanwangi	83	5	415	5,0
6	Kelurahan Blimbing	23	5	115	2,4
7	Kelurahan Purwantoro	82	5	410	4,7
8	Kelurahan Kesatrian	24	5	120	2,6
9	Kelurahan Polehan	54	5	270	4,6
10	Kelurahan Jodipan	34	5	170	3,2
rata-rata				216	3,45

Pada tabel 1, latensi yang terjadi sangat kecil sehingga pada sisi user tidak akan menyebabkan aplikasi jadi lambat. Sehingga penggunaan pemecahan input pengguna per server desa dapat memperkecil latensi sehingga seakan normal seperti sedia kala.

### 3.2 Uji Penggunaan Nonce

Angka nonce berpengaruh pada lama tidaknya pembuatan satu buah blok pada blockchain. Berikut ini adalah waktu yang diperlukan untuk membuat satu blok pada satu kelurahan.

**Tabel 2.** Uji chaining pada tanggal 14 oktober 2020

No	Kelurahan	Jumlah TPS	user	Jumlah blok pada chain	Sub total pembentukan blok(menit)
1	Kelurahan Baleaarjosari	24	5	120	32,5
2	Kelurahan Arjosari	27	5	135	32,6
3	Kelurahan Polowijen	31	5	155	32,8
4	Kelurahan Purwodadi	49	5	245	42,1
5	Kelurahan Pandanwangi	83	5	415	52,0
6	Kelurahan Blimbing	23	5	115	25,4
7	Kelurahan Purwanto	82	5	410	47,7
8	Kelurahan Kesatrian	24	5	120	22,6
9	Kelurahan Polehan	54	5	270	43,6
10	Kelurahan Jodipan	34	5	170	35,2
Rata-rata waktu pembentukan chain per desa atau kelurahan					36,65

Waktu yang diperlukan untuk membuat chain dengan satu digit nol pada semua blok. Contoh sebuah blok dengan digit 0 didepan yaitu :  
 "0519412b0d0055f0f34501cf974878351dd70dd981a91064dc0baf599e42a4ebf2a64268edace2352f4fad9ab17b9659a9d33e68241ba5e0501a81908fa78849". Untuk membuat angka 0 maka kita harus melakukan looping untuk merubah nonce. Hasil acak ini bisa berbeda-beda pada setiap blok. Sehingga hasil total pembentukan blok bisa berbeda-beda pada semua chain.

### KESIMPULAN

Berdasarkan pada uji input http, maka penggunaan server desa terserbar di dalam cluster kubernetes menyebabkan pengguna dari semua TPS akan dikelompokkan koneksinya per desa. Dari total 10 desa dengan rata-rata 216 *concurrent* koneksi menyebabkan latensi pengiriman data dari smartphone petugas ke server yaitu 3,45 menit. Sedangkan waktu pembuatan chain di dalam blockchain untuk semua desa rata-ratanya 37 menit. Sehingga penggunaan ini akan memudahkan pengguna tidak perlu antri di kelurahan/desa. Di satu sisi kepercayaan data dilakukan karena kelambatan proses pembuatan nonce. Untuk penelitian selanjutnya disarankan menambah jumlah digit 0 di depan hash.

### DAFTAR PUSTAKA

- Alam, S. (2019, 5 16). *600 Petugas KPPS Meninggal, MER-C Bentuk Tim Investigasi*. Retrieved 11 06, 2020, from Radio Republik Indonesia CO ID:  
[http://rri.co.id/post/berita/672637/pemilu\\_2019/600\\_petugas\\_kpps\\_meninggal\\_merc\\_bentuk\\_tim\\_investigasi.html](http://rri.co.id/post/berita/672637/pemilu_2019/600_petugas_kpps_meninggal_merc_bentuk_tim_investigasi.html)
- BPS. (2016, Mei 20). *Jumlah Penduduk dan Laju Pertumbuhan Penduduk Menurut Kabupaten/Kota di Provinsi Jawa Timur, 2010, 2014, dan 2015*. Retrieved from Badan Pusat Statistik Provinsi Jawa Timur: <http://jatim.bps.go.id>
- Farisa, F. C. (2019, 7 17). *Kompas Nasional*. Retrieved 10 29, 2020, from KOMPAS:  
<https://nasional.kompas.com/read/2019/07/18/16204331/beda-keterangan-kpu-dan-bawaslu-dalam-sidang-gugatan-nasdem-di-mk?page=all>
- Fontelo, P., & Liu, F. (2013). Mobile App versus Web App: a Comparison Using 2008–2012 “PubMed for Handhelds” Server Data. *AMIA Annual Symposium Proceedings* (pp. 445–450). Bethesda: AMIA.
- KPU. (2019). *HASIL HITUNG SUARA PEMILU PRESIDEN & WAKIL PRESIDEN RI 2019*. Retrieved 11 06, 2020, from Pemilu 2019: <https://pemilu2019.kpu.go.id/#/ppwp/hitung-suara/>
- Nakamoto, S. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System. *Cryptography Mailing list at <https://metzdowd.com>*.
- Satwika, I. K., & Semadi, K. N. (2020). PERBANDINGAN PERFORMANSI WEB SERVER APACHE DAN NGINX DENGAN MENGGUNAKAN IPV6. *SCAN - Jurnal Teknologi Informasi dan Komunikasi*, 15(1), 10-15.

- Warman, I., & Ramdaniansyah, R. (2018). ANALISIS PERBANDINGAN KINERJA QUERY DATABASE MANAGEMENT SYSTEM (DBMS) ANTARA MySQL 5.7.16 DAN MARIADB 10.1. *TEKNOIF*, 6(1), 32-41.
- Yoon, H. J. (2012). A Study on the Performance of Android Platform. *International Journal on Computer Science and Engineering (IJCSE)*, 4(4), 532-537.