

DAFTAR SINGKATAN

UU	=	Undang-undang
API	=	<i>Application Programming Interfaces</i>
IoT	=	<i>Internet of Thing</i>
RFID	=	<i>Radio Frequency Identification</i>
STNK	=	Surat Tanda Nomor Kendaraan
TX	=	Transaksi
RAM	=	<i>Random Access Memory</i>

DAFTAR ISTILAH

- KIR = Merupakan kumpulan rangkaian kegiatan untuk melakukan uji kendaraan bermotor sebagai tanda bahwa kendaraan tersebut layak digunakan secara teknis di jalan raya, khususnya bagi kendaraan yang membawa angkutan penumpang dan barang
- Hardware* = Perangkat keras pada komputer.
- Mobile* = Kata sifat yang berarti dapat bergerak atau dapat digerakkan dengan bebas dan mudah.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pemberitaan mengenai kecelakaan kendaraan umum dan bermuatan belakangan ini cukup marak diberitakan. Mengapa hal yang demikian dapat terjadi? Salah satu penyebab utama dari kecelakaan tersebut adalah adanya kerusakan pada kendaraan itu seperti, rem yang tidak berfungsi, lampu kendaraan yang redup, oli pada kendaraan sudah kering, dan lain-lain. Padahal dalam UU No. 22 Tahun 2009 tentang lalu lintas dan angkutan jalan sudah mengatur standarisasi kelayakan sebuah kendaraan umum dan/atau kendaraan bermuatan, yaitu dengan melakukan tahapan uji KIR. Pengujian ini dilakukan oleh Departemen Perhubungan, keluaran dari uji ini adalah sebuah surat izin. Uji KIR setidaknya harus dilakukan dua kali dalam setahun, sesuai dengan pasal 5 ayat 2 Peraturan Menteri Perhubungan Pengujian Berkala Kendaraan Bermotor (Permenhub PBKB) yang menyatakan uji KIR perdana dilakukan paling lama satu tahun setelah terbit STNK pertama. Perpanjangan uji berskala selanjutnya dilakukan 6 bulan sekali dan dilakukan terus menerus setiap 6 bulan sekali.

Mengapa perlu diadakannya pengujian KIR ini? Salah satu alasannya adalah agar keselamatan pengemudi atau penumpang pada kendaraan tersebut dapat terjamin, sehingga mengurangi angka kematian akibat kecelakaan kendaraan bermotor pada negara ini. Dilansir dari kominfo.go.id “Menurut data kepolisian, di Indonesia rata-rata 3 orang meninggal setiap jam akibat kecelakaan jalan...” [1]. Apakah proses pengujian KIR ini aman dari oknum-oknum yang egois? Tentu saja tidak, seperti dilansir oleh selasar.co “...Hal tersebut diketahui dari pengakuan sopir yang ditemui timnya di tempat pengujian kendaraan itu. ‘Ada yang Rp 225 ribu, ada juga yang Rp 250 ribu. Beda-beda’” [2].

Dari pemaparan masalah di atas, terdapat berbagai solusi untuk mengurangi tindakan pungutan liar tersebut, seperti menerapkan sistem pembayaran secara daring, serta pembukuan atau pencatatan pembayaran melalui kasir. Namun kami

menemukan solusi yang terbaik menurut kami, dikarenakan dengan adanya solusi ini maka proses pengujian KIR dapat berjalan dengan lebih efisien. Solusi yang kami tawarkan adalah dengan melakukan pencatatan pengujian KIR dengan menggunakan sistem *blockchain*. *Blockchain* adalah sebuah *ledger system* yang memungkinkan seseorang melakukan transaksi dengan orang lain tanpa membutuhkan pihak ketiga. Alasan lain kami mengajukan usulan ini dikarenakan penelitian ini adalah pengembangan dari penelitian sebelumnya yaitu tentang “SISTEM PENCATATAN KENDARAAN MENGGUNAKAN BLOCKCHAIN DENGAN BASIS KIR” oleh Andre Dananjaya Kristino. Pada penelitian tersebut menghasilkan sebuah sistem pencatatan kendaraan berbasis KIR dengan menggunakan *blockchain*. Dikarenakan pada penelitian sebelumnya sistem tersebut belum bisa dilakukan secara *mobile* dan terhubung dengan perangkat *hardware*, maka pada penelitian kali ini kami mengusulkan untuk membuat sistem *blockchain* yang sudah dapat terintegrasi ke perangkat *hardware* IoT.

1.2. Rumusan Masalah

Permasalahan yang akan menjadi objek pada penelitian ini adalah:

1. Bagaimana sistem *blockchain* ini dapat mengurangi manipulasi data pada pencatatan uji KIR kendaraan?
2. Bagaimana sistem *blockchain* dapat terintegrasi dengan baik pada perangkat IoT RFID?

1.3. Tujuan dan Manfaat

Berikut merupakan beberapa tujuan yang mendasari dilakukannya penelitian ini, antara lain:

1. Mampu membuat dan mengimplementasikan sebuah sistem *blockchain* yang dapat terintegrasi dengan baik pada perangkat IoT RFID untuk penggunaan validasi serta pembacaan data pencatatan uji KIR kendaraan.

2. Manfaat yang ingin dicapai pada penelitian ini yaitu mengimplementasikan sistem *blockchain* yang dapat terintegrasi dengan baik pada perangkat IoT RFID untuk penggunaan pencatatan hasil uji coba KIR kendaraan.

1.4. Batasan Masalah

Dalam memfokuskan penelitian Tugas Akhir ini, maka diberikan beberapa batasan masalah sebagai berikut:

1. Tempat penelitian hanya dilakukan di Dinas Perhubungan Kabupaten Bandung.
2. Penelitian ini hanya menggunakan *framework hyperledger fabric* versi 2.0 atau lebih, dan tidak membahas tentang *user interface*.
3. Penelitian ini hanya menggunakan sistem *single organization* atau *Private Blockchain* pada sistem *blockchain*-nya.
4. Penelitian ini hanya untuk integritas sistem *blockchain* pada perangkat IoT RFID.
5. Penelitian ini hanya melakukan pengecekan sebelum dilakukannya uji KIR, untuk menyatakan bahwa kendaraan tersebut, sudah lulus uji coba di tahap sebelumnya atau belum.
6. Penelitian ini tidak menyangkut kesalahan data yang disengaja, atau dikarenakan kesalahan manusia (*human error*).

1.5. Metode Penelitian

Pekerjaan penelitian dilakukan dengan pendekatan: studi teoritis/studi literatur, pengukuran empiris, analisis statistik, simulasi, perancangan, dan implementasi.

1.6. Jadwal Pelaksanaan

Berikut adalah rincian kegiatan yang dilakukan pada penelitian ini.

Tabel 1.1 Jadwal dan *milestone*.

No.	Deskripsi Tahapan	Durasi	Tanggal Selesai	<i>Milestone</i>
1	Desain Sistem	3 minggu	6 Juni 2020	Diagram <i>block</i> dan spesifikasi <i>Input-Output</i>
2	Pemilihan Komponen Server dan Bahasa Pemrograman yang digunakan	2 minggu	21 Juni 2020	<i>List</i> komponen yang akan digunakan
3	Pembuatan server	2 minggu	5 Juli 2020	<i>Nodes</i> 1 dan <i>nodes</i> 2 selesai
4	Penulisan program API	5 bulan	5 Desember 2020	<i>Prototype</i> 1 selesai
5	Penyempurnaan program API	2 minggu	18 Februari 2021	<i>Prototype</i> 2 selesai
6	Pembuatan tampilan <i>dashboard</i>	3 minggu	12 Maret 2021	<i>Dashboard</i> selesai
7	Pengujian fungsi	2 minggu	23 April 2021	<i>Functionality testing</i>
8	Penyusunan Buku TA	2 minggu	7 Mei 2021	Buku TA selesai

BAB II

TINJAUAN PUSTAKA

2.1. *Blockchain*

Blockchain awalnya diperkenalkan oleh seseorang (atau sekelompok orang) yang dikenal sebagai Satoshi Nakamoto pada tahun 2008, berupa sebuah rancangan sistem [3]. Kemudian dikembangkan dan diimplementasikan pada tahun 2009 [4]. Beberapa tahun terakhir, *blockchain* mendapat perhatian besar oleh dunia perindustrian baik dari sektor keuangan, kesehatan, pemerintahan, kegunaan, dan pendidikan. Alasan terbesar kenapa *blockchain* ini dapat berkembang dengan sangat pesat serta menarik minat banyak orang adalah dengan *blockchain* ini, sebuah aplikasi yang dulunya hanya dapat berjalan dengan perantara yang dipercaya dan bersifat *centralized*, sekarang dapat berjalan secara *decentralized* tanpa diperlukannya *central authority*, dan dengan fungsi serta manfaat yang sama, yang sebelumnya ini adalah hal yang tidak mungkin [5].

Blockchain adalah *distributed database* yang merekam semua transaksi yang terjadi di dalam sebuah *network* [6]. Maksud dari *distributed database* disini adalah sebuah catatan (*record*) dari sebuah data yang disimpan secara menyeluruh oleh seluruh penggunanya. Sistem terdistribusi ini disebut sebagai *distributed ledger*. Dari sisi komputasi, *blockchain* adalah struktur data *entries* yang tersimpan terhubung satu dengan yang lainnya dengan urutan yang tepat [7]. Pada dasarnya *blockchain* memiliki 4 pilar utama yaitu: *Consensus*, adalah penyedia pembuktian hasil kerja dari *blockchain* dan sebagai bentuk verifikasi *action* pada *network*; *Ledger*, adalah penyedia detail lengkap mengenai sebuah transaksi yang terdapat pada sebuah *network*; *Cryptography*, yang memastikan semua data yang tersimpan pada *ledger* atau *network* di-enkripsi, sehingga data tersebut hanya bisa diakses oleh *authorized user*; *Smart Contracts*, yang mana berguna untuk melakukan verifikasi kepada semua *participants* dari sebuah *network* [8].

Menurut Stefen K, *blockchain* dibagi menjadi 3 jenis, yaitu: *Private Blockchain*, *Public Blockchain*, dan *Consortium Blockchain*. *Public blockchain*

adalah *blockchain* dengan *network* yang dapat diakses oleh siapapun, serta siapapun dapat melakukan transaksi di dalamnya. *Private Blockchain* adalah *blockchain* yang hanya dapat diakses oleh *authorized user* dan yang dapat melakukan transaksi adalah pengguna yang sudah ditetapkan. *Consortium blockchain* adalah *blockchain* yang dapat diakses oleh beberapa organisasi yang sudah ditetapkan [6]. Untuk lebih jelasnya perhatikan Tabel 2.1 berikut.

Tabel 2.1 Perbedaan jenis-jenis *blockchain*.

<i>Property</i>	<i>Public Blockchain</i>	<i>Consortium Blockchain</i>	<i>Private Blockchain</i>
<i>Consensus Determination</i>	<i>All Miners</i>	<i>The selected set of Nodes</i>	<i>One organization</i>
<i>Read Permission</i>	<i>Public</i>	<i>Could be public or restricted</i>	<i>Could be public or restricted</i>
<i>Immutability</i>	<i>Nearly impossible to tamper</i>	<i>Could be tampered</i>	<i>Could be tampered</i>
<i>Efficiency</i>	<i>LOW</i>	<i>HIGH</i>	<i>HIGH</i>
<i>Centralized</i>	<i>No</i>	<i>Partial</i>	<i>Yes</i>
<i>Consensus Process</i>	<i>Permission less</i>	<i>Permissioned</i>	<i>Permissioned</i>

2.2. Karakteristik dari *Blockchain*

Blockchain memiliki beberapa karakteristik yang membedakannya dengan *database*, yaitu [4] [9]:

1. *Decentralization*. Pada sistem yang *centralized* diperlukannya pihak ketiga sebagai penengah dan validator untuk setiap transaksinya. Proses ini akan memakan waktu yang cukup lama serta pengeluaran yang sangat mahal. Pada *blockchain* sebuah transaksi dapat dilakukan tanpa adanya bantuan dari pihak ketiga, dikarenakan *blockchain* menggunakan *consensus algorithm*-nya untuk mempertahankan keaslian dari sebuah transaksi.

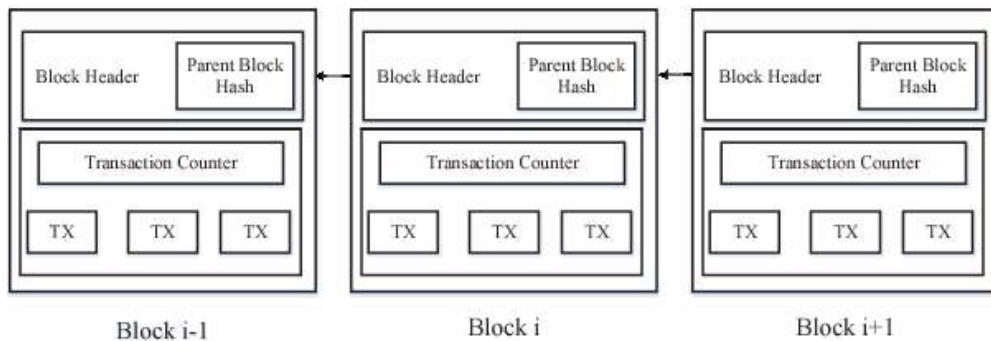
2. *Persistency*. Sebuah transaksi dapat di validasi dengan sangat cepat, dan setiap transaksi yang tidak valid akan langsung di tolak oleh *participants*. Sehingga setiap transaksi yang tersimpan pada *blockchain* sangat mustahil untuk di-*rollback* ataupun dihapus.

3. *Anonymity*. Setiap pengguna dapat berinteraksi dengan *blockchain* dengan sebuah alamat yang tidak menampilkan alamat asli dari pengguna tersebut.

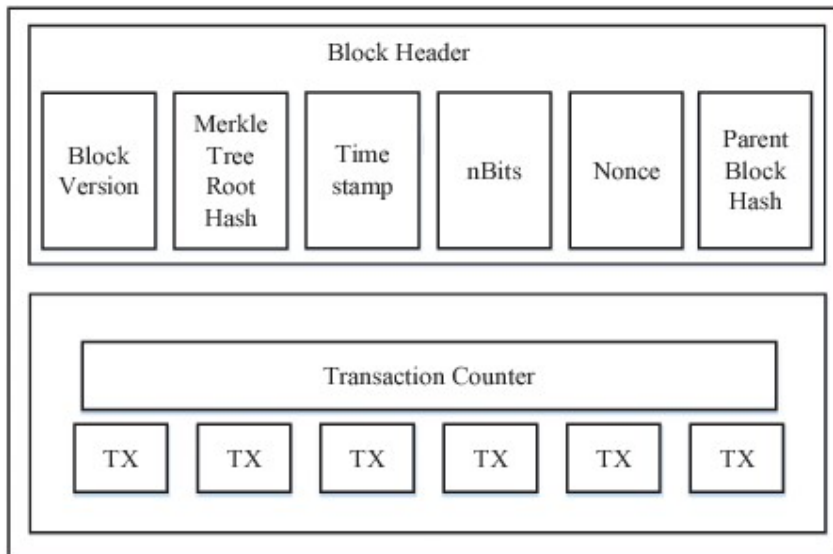
4. *Auditability*. Setiap transaksi yang tercatat dalam *blockchain* akan saling terhubung antar satu dengan yang lainnya, sehingga memudahkan dalam melakukan *tracking* data.

2.3. *Block Architecture*

Pada Gambar 2.1 digambarkan struktur dari sebuah sistem *blockchain* yang saling terhubung. Pada umumnya struktur dari *blockchain* adalah susunan dari beberapa *block* yang saling terhubung. Setiap *block* memiliki nilai *hash*-nya masing-masing yang terhubung dengan nilai *hash* dari *block* sebelumnya. Terdapat pengecualian pada *block* awal sebuah *blockchain* yang tidak memiliki nilai *parent hash*, *block* ini biasa disebut dengan *genesis block* [4].



Gambar 2.1 *Blockchain* yang terdiri dari beberapa *block* yang saling terhubung.



Gambar 2.2 Bagian-bagian yang terdapat pada sebuah *block*.

Seperti yang dilihat pada Gambar 2.2 bahwa pada sebuah *block* terdapat 2 bagian inti yaitu *block header* dan *block body*. Pada *block header* terdapat beberapa bagian diantaranya [4]:

1. *Block version*. Menunjukkan *rules block* mana yang harus diikuti .
2. *Merkle tree root hash*. Nilai *hash* dari semua transaksi yang terdapat dalam sebuah *block*.
3. *Timestamp*. Waktu sekarang dalam hitungan detik dengan format hitungan universal dimulai dari 1 Januari 1970.
4. *nBits*. Target *thresholds* untuk proses validasi *hash* pada *block*.
5. *Nonce*. Adalah 4-byte data yang diawali dengan 0 serta meningkat setiap perhitungan nilai *hash*.
6. *Parent block hash*. Nilai *hash* dengan panjang 256-byte yang berhubungan dengan *parent block hash* sebelumnya.

Sedangkan pada *block body* berisikan jumlah transaksi yang terdapat dalam *block* serta isi dari transaksi itu sendiri.

2.4. Cara Kerja *Blockchain*

Blockchain adalah struktur data yang terdistribusi dan tersebar di seluruh komputer pada sebuah *network*. Pertama kali diperkenalkan pada pengaplikasian oleh Bitcoin untuk mengatasi masalah *double spending*. Sebagai hasilnya bagaimana *nodes* pada Bitcoin *network* (biasanya disebut *miners*) melakukan validasi dan melakukan transaksi yang sudah disepakati bersama, *blockchain* pada Bitcoin menentukan kepemilikan dari siapa memiliki apa. Berikut adalah gambaran bagaimana *blockchain* bekerja pada sebuah *peer-to-peer network* [5].

1. Pengguna berinteraksi dengan *blockchain* menggunakan *asynchronous encryption key (Private dan Public key)*. Pengguna menggunakan *Private Key* mereka untuk menandai transaksi yang dibuat oleh pengguna itu sendiri, dan menuliskan alamatnya ke dalam *blockchain* dengan menggunakan *Public Key* mereka. Setiap alamat transaksi akan di-*broadcast* kepada seluruh pengguna yang berada *one-hop peers* didekatnya.
2. *Peer* tetangga yang menerima transaksi baru ini akan melakukan validasi sebelum dilanjutkan ke dalam *blockchain*. Transaksi yang tidak valid akan langsung dibuang oleh *peer* kembali ke *peer* awal.
3. Transaksi yang sudah dikumpulkan dan divalidasi, akan dikemas menjadi sebuah *timestamp candidate block*. Ini adalah proses yang disebut *mining*. *Mining node* akan meneruskan paket ini kedalam *network*.
4. *Network nodes* akan melakukan verifikasi terhadap *block* tadi (a) memiliki transaksi yang valid, (b) nilai *hash*-nya berhubungan dengan nilai *hash* pada *block* sebelumnya. Jika semuanya valid maka *block* tersebut akan ditambahkan pada *chain* yang ada.

Kapan sebuah transaksi dikatakan sebagai sebuah transaksi yang valid? Perlu diingat bahwa pada jaringan *blockchain* sebenarnya kita hanya memiliki sekumpulan kode yang ditulis oleh orang yang tidak kita percaya pada sebuah *shared database* tanpa adanya pihak ketiga, maka diperlukannya sebuah aturan yang harus disetujui oleh seluruh *nodes* yang terhubung dalam jaringan tersebut,

dan aturan ini harus disebar dan harus ada pada setiap *nodes*, sehingga setiap transaksi yang masuk akan menghasilkan keluaran yang sama, walaupun jenis transaksinya berbeda.

2.5. *Smart Contract*

Smart contract adalah sekumpulan Respon-Skenario yang berisikan prosedur atau aturan, dan logika [10]. Dengan kata lain *smart contract* adalah *distributed application* yang terdapat dalam sebuah *blockchain*, yang ditulis dengan bahasa pemrograman tertentu serta berisikan *class* dan *method*, setiap pengguna dapat memanggil *class* dan *method* ini dengan mengirimkan transaksi ke “alamat” yang terdapat pada *blockchain*. Setiap kali pengguna melakukan sebuah transaksi maka *smart contract* akan secara otomatis dijalankan di semua *nodes* yang terdapat pada jaringan *blockchain* dengan derismatik dan *reliable* [11]. Setiap anggota atau *nodes* akan menandatangani kontrak ini secara bersama, serta menyimpannya sendiri, nantinya proses penandatanganan ini akan dilakukan secara otomatis oleh setiap *nodes* atau anggota, sebagai sebuah bentuk kesepakatan bersama dapat tercapai dalam proses validasi sebuah transaksi [10]. Sebagai catatan *smart contract* ini adalah kode yang bersifat *immutable*, walaupun oleh pembuatnya sendiri, dan tidak berkepemilikan.

Smart contract memiliki 3 karakteristik utama yaitu: *autonomy*, *self-sufficiency*, dan *decentralization*. Autonomi disini adalah ketika kontrak dijalankan, sang pemilik kontrak dan agen-agensya tidak perlu terhubung secara langsung. Untuk *self-sufficiency* sendiri berarti *smart contract* dapat mengembangkan sendiri kemampuan mereka dalam mengelola sumber daya seperti mengumpulkan data, menyediakan layanan, dan mengeluarkan suatu data jika dibutuhkan. Terakhir *decentralization* maksudnya adalah *smart contract* tidak tersimpan dalam suatu server yang terpusat, mereka didistribusikan kepada seluruh *nodes* pada *network* [10].

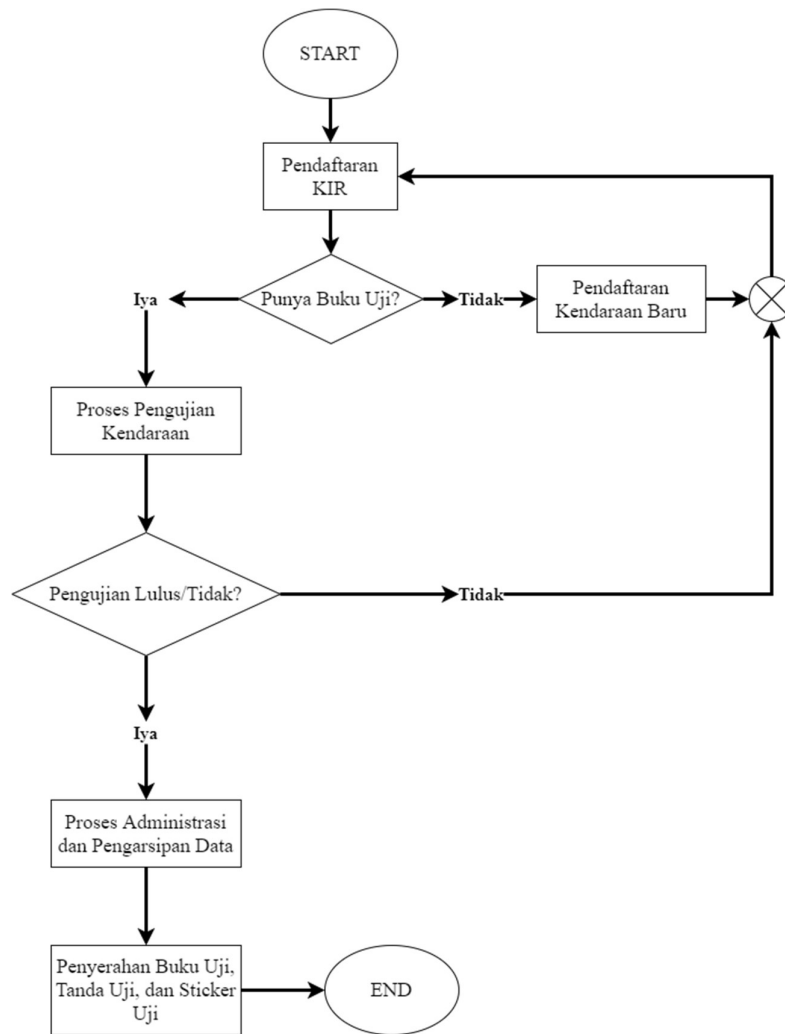
2.6. *Internet of Things (IoT)*

Internet of Things (IoT) adalah sebuah objek yang digunakan untuk menghubungkan berbagai *devices* pada sebuah jaringan untuk mengambil data dari internet yang digunakan dalam berbagai perangkat cerdas dan terhubung dengan *embedded system* seperti, *software*, sensor-sensor, dan kecerdasan buatan [9]. *Internet of Things (IoT)* sekarang ini menjadi batu loncatan untuk pengembangan dunia dari *real world entities* menuju *intelligent world entities* [12]. Sejauh ini teknologi IoT sudah berkembang dengan sangat luas, dan sebagian besar perusahaan sedang mengembangkan *machine-to-machine (M2M) communication* [13], yang memungkinkan setiap perangkat yang terhubung dengan jaringan internet dapat melakukan komunikasi dan saling bertukar informasi.

BAB III ANALISIS DAN PERANCANGAN

3.1. Desain Sistem

Pada bab 3 ini dijelaskan, tentang gambaran umum desain sistem yang dibuat. Berikut penjelasan mengenai proses KIR SOP 35 Kabupaten Bandung.



Gambar 3.1 Flowchart dari proses KIR yang sekarang digunakan.

Pada gambar 3.1 adalah diagram alir atau *flowchart* proses KIR, di mana diagram tersebut dibuat berdasarkan SOP pelayan pengujian berkala kendaraan bermotor Dinas Perhubungan Kabupaten Bandung. Prosesnya adalah pemohon melakukan pendaftaran kendaraan dengan menyerahkan buku uji, fotokopi STNK,

fotokopi KTP, pembayaran retribusi, penyerahan super uji dan tester, dengan membawa kendaraan. Jika pemohon belum memiliki buku uji atau dapat dikatakan kendaraan yang di uji masih baru maka pemohon melakukan pendaftaran kendaraan baru, setelah itu dapat melakukan proses pendaftaran. Di mana untuk proses pendaftaran memakan waktu kurang lebih 4,5 menit.

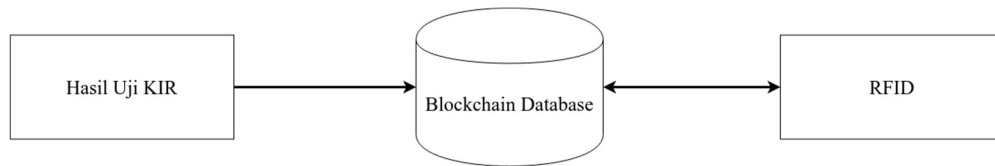
Setelah proses pendaftaran selesai pemohon akan mendapatkan blangko surat uji yang nantinya akan diserahkan kepada penguji untuk proses pengujian, dari mulai tes visual hingga tes fisik kendaraan. pada proses pengujian memakan waktu sekitar 15,5 menit. Jika kendaraan dinyatakan tidak lulus dikarenakan ada kendala teknis, maka kendaraan diwajibkan untuk memperbaiki kendaraan tersebut. Kerusakan tersebut dibagi menjadi 2 yaitu kerusakan ringan dan kerusakan berat, di mana untuk kerusakan ringan akan diberi waktu untuk perbaikan 1 hari sedangkan untuk kerusakan berat diberi waktu 1 minggu. Jika dalam kurun waktu tersebut pemohon dapat memperbaikinya pemohon tidak perlu membayar biaya retribusi lagi dan melakukan KIR ulang, sebaliknya jika pemohon tidak dapat memperbaikinya pemohon diwajibkan untuk membayar biaya retribusi lagi dan melakukan KIR ulang.

Setelah proses pengujian berhasil pemohon memberikan kuitansi biaya retribusi ke loket. Penguji akan menyerahkan data dari hasil pengujian ke petugas loket, petugas loket melakukan pengarsipan dari hasil pengujian dan kuitansi pembayaran pemohon. Setelah itu petugas loket akan menandatangani buku uji, stiker uji, dan tanda uji. Waktu yang diperlukan sekitar 24 menit, setelah semua proses selesai pemohon akan mendapatkan buku uji, stiker uji baru, dan tanda uji baru. Total waktu yang diperlukan untuk proses KIR adalah 44 menit.

3.1.1. Diagram Blok

Secara garis besar sistem yang akan dibuat adalah, perangkat IoT RFID *reader* akan membaca RFID yang sudah ditempelkan pada kendaraan, dan mengirimkan ID dari RFID tersebut ke dalam *blockchain database*, jika ID tersebut tersedia maka *blockchain* akan mengirimkan sinyal kepada RFID untuk menyatakan bahwa data dari RFID tersebut tersedia dan dapat melakukan tahapan uji KIR. Jika ID dari RFID itu tidak tersedia maka pemilik kendaraan diharapkan

untuk melakukan pendaftaran terlebih dahulu pada bagian Administrasi. Hasil uji KIR nanti akan dikirimkan ke dalam *blockchain database* untuk dilakukan verifikasi dan *update* data. Dikarenakan uji KIR terdapat beberapa tahap, maka setiap kendaraan selesai melakukan uji KIR pada satu tahap, untuk dapat melanjutkan ke tahapan selanjutnya, RFID dari kendaraan tersebut akan dibaca ulang oleh RFID *reader* untuk memastikan bahwa kendaraan tersebut sudah lulus uji tahap tersebut. Jika kendaraan dinyatakan lulus tahap tersebut, maka kendaraan dapat melanjutkan ke tahap selanjutnya. Sedangkan, jika kendaraan tidak lulus dari tahapan tersebut, maka pemilik kendaraan diminta untuk melakukan perbaikan, dan proses uji KIR akan berakhir. Untuk lebih jelasnya dapat memperhatikan Gambar 3.2.



Gambar 3.2 Diagram blok dari rancangan sistem.

Untuk mengatasi masalah duplikasi data maka pemetaan ID untuk setiap datanya pada *blockchain* akan menggunakan ID dari RFID *tag* yang terdapat pada kendaraan tersebut. Jika pemilik kendaraan mengalami masalah seperti tidak lulus pada tahapan uji KIR atau ID dari RFID kendaraan tidak terdaftar pada *blockchain database*, maka pemilik kendaraan tersebut akan diminta untuk melengkapi segala kekurangannya di bagian Administrasi, dan jika kendaraan tersebut sebelumnya tidak lulus dalam salah satu tahapan uji KIR, maka kendaraan tersebut harus mengulang dari tahapan awal pengujian, dan data uji yang sebelumnya akan digantikan atau diisikan oleh sistem menjadi *null* dan dinyatakan tidak lulus uji KIR. Untuk lebih jelas dalam pemahaman alur sistem yang akan dibuat, dapat memperhatikan Gambar 3.3.