

		No.FO.11.3.3-V3	HAL. 1/32
PPPM	DIR	Format PPPM: Laporan Kemajuan Penelitian/Pengabdian Kepada Masyarakat	
10 Februari 2020			

LAPORAN AKHIR
Penelitian Terapan



PENERAPAN BLOCKCHAIN PADA DIGITAL SIGNATURE QR CODE

Tahun ke 1 dari rencana 1 tahun

Ketua/Anggota Tim

Supardianto, S.ST., M.Eng – 113105

Hamdani Arif, S.Pd., M.Sc. – 117175

Maidel Fani, S.Pd., M.Kom. – 117192

Hajrul Khaira, S.Tr.Kom – 220315

Syeirin Armanda Syettia – NIM: 3312101034

Tunas Safetyan – NIM: 3312101061

Yogi Rahman Alif - NIM: 4332201012

Fathur Wiriansyah – NIM: 4332201023

Alfaturrahman – NIM: 4342201045

POLITEKNIK NEGERI BATAM

Desember 2023

HALAMAN PENGESAHAN

Judul : Penerapan Blockchain pada Digital Signature QR Code
Peneliti / Pelaksana :
Nama Lengkap : Supardianto, S.ST., M.Eng.
NIK/NIDN : 113105 / 0020049008
Jabatan Fungsional : Asisten Ahli
Program Studi : D4 TEKNOLOGI REKAYASA PERANGKAT LUNAK
Nomor HP : 085762315262
Alamat surel (e-mail) : Supardianto@polibatam.ac.id
Anggota (1)
Nama Lengkap : Hamdani Arif, S.Pd., M.Sc
NIK/NIDN : 117175 / 0001129002
Program Studi : D4 REKAYASA KEAMANAN SIBER
Anggota (3)
Nama Lengkap : Maidel Fani, S.Pd., M.Kom.
NIK/NIDN : 117192 / 0008058807
Program Studi : D4 REKAYASA KEAMANAN SIBER
Anggota (4)
Nama Lengkap : Hajrul Khaira, S.Tr.Kom
NIK/NIDN : 220315 /
Program Studi : D4 REKAYASA KEAMANAN SIBER
Insitusi Mitra (jika ada)
Nama Insitusi Mitra :
Alamat :
Penganggung Jawab :
Tahun Pelaksanaan : Tahun ke 1 dari rencana 1 tahun
Biaya Tahun Berjalan : Rp. 13.375.000
Biaya Keseluruhan : Rp. 13.375.000

Batam, 01 Desember 2023

Mengetahui
Ketua CoX
Polibatam Software Team
Ketua,



(Metta Santiputri, S.T., M.Sc, Ph.D.)
NIK/NIDN. 100017 / 1020077701

A handwritten signature in black ink, consisting of several loops and strokes.

(Supardianto, S.ST., M.Eng.)
NIK/NIDN. 113105 / 0020049008

RINGKASAN

Digital signature berbasis QR code telah menjadi metode yang banyak digunakan dalam verifikasi dan otentikasi dokumen elektronik dan sudah diterapkan pada Jurusan Teknik Informatika. Namun, sistem ini masih memiliki kelemahan dalam hal keamanan dan keabsahan data. Risiko manipulasi dan pemalsuan data serta adanya potensi penggantian QR code dapat mengurangi integritas dari QR code yang dibuat. Teknologi Blockchain menawarkan potensi untuk mengatasi permasalahan ini. Blockchain menyediakan sistem terdistribusi yang aman dan transparan dengan mekanisme konsensus yang menjaga integritas data. Penerapan teknologi Blockchain pada sistem digital signature berbasis QR code dapat meningkatkan keamanan dan keabsahan data QR code yang dibuat. Namun, penerapan teknologi saat ini digunakan masih terbatas dan memiliki kekurangan. Oleh karena itu, penelitian ini bertujuan untuk menerapkan Blockchain pada sistem digital signature berbasis QR code yang telah digunakan oleh Jurusan Teknik Informatika. Dengan menggabungkan kepraktisan QR code dan keamanan Blockchain, penelitian ini diharapkan dapat meningkatkan keamanan, keabsahan, dan integritas dokumen elektronik yang ditandatangani menggunakan digital signature berbasis QR code yang telah dibuat. Hasil penelitian ini juga dapat menjadi dasar pengembangan lebih lanjut dalam meningkatkan sistem digital signature berbasis QR code secara umum, dengan berbagai potensi aplikasi di bidang keuangan, logistik, dan administrasi. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi penting dalam mengatasi kelemahan sistem digital signature berbasis QR code saat ini dan membuka peluang baru untuk pemanfaatan teknologi Blockchain dalam meningkatkan keamanan informasi yang ditandatangani menggunakan QR code.

Kata Kunci : *Blockchain; QR code; Keabsahan data*

PRAKATA

Puji syukur dan alhamdulillah senantiasa penulis panjatkan ke hadirat Allah S.W.T yang telah melimpahkan Rahmat serta Petunjuk-Nya, sehingga kegiatan penelitian dosen muda ini dapat diselesaikan. Penelitian ini mengambil judul: **PENERAPAN BLOCKCHAIN PADA DIGITAL SIGNATURE QR CODE**. Beberapa pihak telah memberikan bantuan dalam rangka penyelesaian laporan ini. Oleh karena itu dalam kesempatan ini penulis ingin menyampaikan penghargaan dan ucapan terimakasih yang sebesar-besarnya kepada pihak-pihak tersebut, yaitu:

1. Ketua P3M dan Staff di Politeknik negeri Batam
2. Anggota Tim dan kerjasama tim dalam kegiatan ini
3. Seluruh civitas akademika Politeknik Negeri Batam yang sudah berkontribusi

Penulis menyadari bahwa hal yang disajikan dalam laporan ini masih banyak kekurangannya. Oleh karena itu saran dan kritik yang bersifat membangun sangat diharapkan.

Batam, 01 Desember 2023

Tim Peneliti

DAFTAR ISI

HALAMAN SAMPUL	1
HALAMAN PENGESAHAN	2
RINGKASAN	3
PRAKATA	4
DAFTAR ISI	5
BAB 1. PENDAHULUAN	6
BAB 2. TINJAUAN PUSTAKA	7
BAB 3. TUJUAN DAN MANFAAT PENELITIAN	9
BAB 4. METODE PENELITIAN	10
BAB 5. HASIL YANG DICAPAI	13
BAB 6. KESIMPULAN DAN SARAN	16
DAFTAR PUSTAKA	17
LAMPIRAN	

BAB 1. PENDAHULUAN

Penerapan tanda tangan digital atau biasa dikenal dengan digital signature saat ini sudah mulai umum digunakan. Penerapan dengan memanfaatkan QR Code menjadi metode yang populer, karena dapat melakukan verifikasi dan otentifikasi pada dokumen yang disertai QR Code tersebut. Di Politeknik Negeri Batam, khusus nya di Jurusan Teknik Informatika sudah mulai menerapkan metode tersebut pada kegiatan yang memerlukan tanda tangan oleh dosen, seperti persetujuan laporan, serta perihal izin, dll. Namun, sistem yang dikembangkan saat ini masih memiliki kelemahan dalam hal keamanan dan keabsahan data yang ditandatangani. Risiko manipulasi dan pemalsuan data serta serangan penggantian QR code menjadi salah satu ancaman dalam penerapannya.

Solusi yang coba digunakan untuk mengatasi masalah ini adalah dengan menggunakan teknologi Blockchain. Teknologi blockchain menyediakan sistem terdistribusi yang aman dan transparan dengan mekanisme konsensus yang menjaga integritas data. Penerapan teknologi Blockchain pada sistem digital signature berbasis QR code dapat meningkatkan keamanan dan keabsahan data yang ditandatangani.

Dengan menggabungkan kepraktisan QR code dan keamanan Blockchain, penelitian ini diharapkan dapat meningkatkan keamanan, keabsahan, dan integritas dokumen elektronik yang ditandatangani menggunakan digital signature berbasis QR code. Hasil penelitian ini juga dapat menjadi dasar pengembangan lebih lanjut dalam meningkatkan sistem digital signature berbasis QR code secara umum, dengan berbagai potensi aplikasi di bidang keuangan, logistik, dan administrasi.

Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi penting dalam mengatasi kelemahan sistem digital signature berbasis QR code saat ini dan membuka peluang baru untuk pemanfaatan teknologi Blockchain dalam meningkatkan keamanan informasi yang ditandatangani menggunakan QR code.

Penelitian ini juga masuk dalam pengerjaan project based learning di program studi Rekayasa Keamanan Siber, dengan melibatkan beberapa matakuliah seperti Administrasi sistem komputer, hukum dan etika siber, interkoneksi jaringan, keamanan basis data, kriptografi terapan, dan pemrograman berorientasi objek. Luaran yang diharapkan dari penelitian ini adalah publikasi pada jurnal nasional dan diharapkan produk yang dihasilkan bisa di HKI kan.

BAB 2. TINJAUAN PUSTAKA

Blockchain, sebagai teknologi yang memungkinkan pencatatan data transaksional dalam format yang aman dan tidak dapat diubah, telah mendapatkan banyak perhatian dalam beberapa dekade terakhir. Seperti yang dijelaskan oleh Narayanan et al. (2016), blockchain adalah rantai blok yang saling terhubung dan tersertifikasi dengan enkripsi. Setiap blok memuat informasi transaksi, sebuah timestamp, dan referensi ke blok sebelumnya. Keunggulan utama dari teknologi ini, seperti yang ditunjukkan oleh Zheng et al. (2018), adalah sifatnya yang tahan terhadap modifikasi data. Sekali data direkam di dalam blockchain, data tersebut hampir mustahil untuk diubah tanpa perubahan pada seluruh rantai.

Sementara itu, tanda tangan digital merupakan salah satu teknologi yang paling penting dalam verifikasi asal-usul dan integritas dokumen digital. Rivest et al. (1978) menggambarkan digital signature sebagai teknik yang mengenkripsi informasi dengan menggunakan kunci pribadi pemilik tanda tangan. Keamanan yang diberikan teknologi ini memungkinkan penerima untuk memastikan bahwa dokumen yang diterima berasal dari pengirim yang diklaim dan tidak diubah selama proses pengiriman (Housley et al., 2002).

Di sisi lain, QR Code, seperti yang digambarkan oleh Denso Wave Incorporated (1994), adalah bentuk evolusi dari kode batang tradisional, yang dapat menyimpan informasi dalam bentuk teks, URL, gambar, dan lainnya. QR Code memiliki kapasitas penyimpanan yang lebih besar dan dapat dengan cepat discan oleh perangkat mobile, menjadikannya pilihan populer dalam bidang iklan, pembayaran mobile, dan otentikasi (Kirov, 2012).

Ketika kita menggabungkan ketiga teknologi ini, terdapat peluang untuk menciptakan sistem yang lebih aman, transparan, dan efisien. Dengan mengintegrasikan blockchain dengan digital signature dalam QR Code, kita dapat meningkatkan keamanan dan otentikasi data yang disimpan dalam QR Code. Sebagaimana ditunjukkan oleh Zohar (2015), sistem semacam ini memastikan bahwa informasi yang dikandung dalam QR Code tidak hanya otentik, tetapi juga tidak dapat diubah-ubah. Tapscott & Tapscott (2016) juga menyoroti bagaimana keunikan dari blockchain dapat menambah tingkat transparansi dan memungkinkan pelacakan setiap perubahan data dengan jelas.

Usulan penelitian ini memang memiliki potensi yang signifikan dalam mengatasi tantangan tanda tangan digital berbasis QR code melalui pemanfaatan teknologi blockchain. Untuk memberikan gambaran yang lebih rinci mengenai metodologi dan implikasi praktis dari solusi yang diusulkan, kita dapat merinci beberapa aspek kunci sebagai berikut:

- **Proses Verifikasi Tanda Tangan:**
Solusi ini dapat melibatkan proses verifikasi tanda tangan digital dengan menggabungkan keunggulan tanda tangan digital berbasis QR code dengan keamanan dan keandalan teknologi blockchain. Tanda tangan digital dapat diintegrasikan ke dalam transaksi blockchain dengan metode kriptografi yang kuat, sehingga transaksi dapat diverifikasi dan divalidasi secara aman.
- **Pemastian Integritas Data:**
Teknologi blockchain dapat digunakan untuk memastikan integritas data dengan menyimpan hash atau tanda identifikasi unik dari setiap transaksi atau dokumen. Setiap perubahan atau manipulasi pada data akan terdeteksi dengan jelas melalui perubahan pada hash, sehingga memastikan bahwa data tetap tidak berubah dan terjamin keasliannya.
- **Mekanisme Konsensus:**
Konsensus dalam konteks blockchain biasanya merujuk pada proses di mana seluruh jaringan setuju tentang versi data yang benar. Ini adalah elemen kunci dalam banyak sistem blockchain seperti Bitcoin, di mana konsensus diperoleh melalui mekanisme Proof-of-Work. Jika ingin menggunakan relational maka struktur data blockchain merupakan implementasi blockchain yang terpusat, di mana konsensus mungkin tidak diperlukan dalam konteks yang sama dengan sistem blockchain yang terdesentralisasi.
- **Metode Penelitian dan Evaluasi:**
Dalam menguji dan mengevaluasi efektivitas solusi yang diusulkan, Anda dapat mengusulkan pendekatan pengujian simulasi atau pengujian nyata di lingkungan yang sesuai. Jelaskan bagaimana Anda akan melakukan pengujian transaksi, verifikasi tanda tangan, serta pemastian integritas data pada jaringan blockchain. Evaluasi juga bisa mencakup analisis performa, kecepatan transaksi, dan tingkat keamanan yang dicapai oleh solusi ini.
- **Implikasi Praktis:**
Penting untuk menjelaskan bagaimana solusi ini akan memberikan dampak praktis pada pengguna. Misalnya, bagaimana UMKM atau pengguna lainnya akan mengadopsi teknologi ini dalam kehidupan sehari-hari mereka? Bagaimana penggunaan teknologi blockchain dapat mengurangi risiko pemalsuan atau manipulasi data, dan bagaimana solusi ini akan memberikan manfaat yang lebih besar daripada solusi konvensional?

Dengan memberikan penjelasan mendalam tentang aspek-aspek di atas, usulan penelitian ini akan lebih kuat dan dapat memberikan pemahaman yang lebih komprehensif kepada pembaca mengenai bagaimana solusi yang diusulkan akan diimplementasikan, diuji, dan memberikan dampak praktis yang signifikan.

BAB 3. TUJUAN DAN MANFAAT PENELITIAN

3.1 Tujuan Penelitian

Penelitian yang diusulkan memiliki tujuan utama untuk mengintegrasikan teknologi blockchain ke dalam tanda tangan digital berbasis QR Code. Motivasi dari integrasi ini adalah meningkatnya kebutuhan akan solusi keamanan yang lebih tangguh dan handal di era digital saat ini. Digital signature berbasis QR Code sudah dikenal memiliki potensi yang baik dalam memastikan integritas dan keaslian informasi. Namun, dengan menggabungkannya dengan keunggulan teknologi blockchain, diharapkan akan ada peningkatan signifikan dalam hal keamanan dan keabsahan data.

3.2 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1. Meningkatkan tingkat keamanan pada tanda tangan digital berbasis QR Code yang telah diaplikasikan sebelumnya.
2. Meningkatkan kepercayaan terhadap penggunaan tanda tangan digital.
3. Menambahkan fokus penelitian pada bidang blockchain.

BAB 4. METODE PENELITIAN

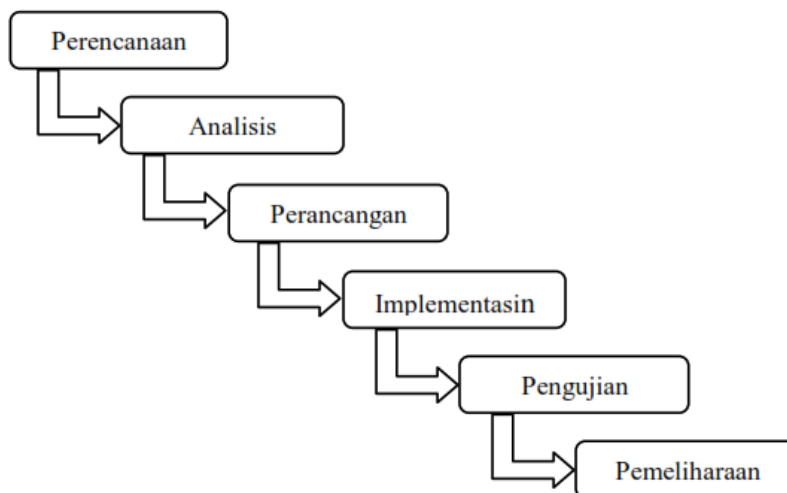
4.1 Pengumpulan Data

Melakukan pengumpulan data dari sumber yang berkaitan dengan perancangan system. Pengumpulan data melalui survei pada tempat penelitian, data yang dikumpulkan adalah dokumen berformat (txt, docx, dan pdf). Data yang telah dikumpulkan kemudian dianalisis sesuai kebutuhan dalam melakukan perancangan.

4.2 Perancangan Sistem

SDLC merupakan kependekan dari System Development Life Cicle yang artinya siklus atau tahapan pengembangan perangkat lunak. Model pengembangan SDLC ini sendiri ada begitu banyak, namun yang ada sejak awal dan masih digunakan sampai dengan sekarang, serta terkenal kepopulerannya, salah satunya adalah model Waterfall (Ruparelia, 2010; S, 2017) . Oleh karena itu penelitian ini menggunakan metode pengembangan perangkat lunak SDLC dengan model Waterfall ini. Model pengembangan perangkat lunak ini terdiri dari tahapan analisis (analysis), perancangan (design), implementasi (implementation), pengujian (testing), dan diseminasi (deployment). Tahapan-tahapan ini harus diselesaikan terlebih dahulu satu demi satu, dimana perpindahan ke tahapan berikutnya hanya bisa dilakukan jika tahapan sebelumnya benar-benar telah selesai. Untuk itu, model Waterfall juga bersifat rekursif, yang artinya setiap tahapan dapat diulang sampai tahapan tersebut benar-benar sempurna (Bassil, 2015). Adapun lebih jelasnya, tahapan/alur proses model Waterfall ini dapat dilihat pada Gambar 3.1 di bawah ini.

Tahapan dari model waterfall adalah sebagai berikut:



- Tahap Perencanaan
Tahap ini menyangkut studi tentang kebutuhan pengguna (user Specification), studi-studi kelayakan baik secara teknis maupun secara teknologi, serta penjadwalan pengembangan suatu proyek sistem informasi dan perangkat lunak.
- Tahap Analisis
Tahap untuk berusaha mengenali permasalahan yang muncul pada pengguna, mengenali komponen-komponen sistem atau perangkat lunak, objek-objek, hubungan antara objek dan sebagainya.
- Tahap Perancangan
Pada tahap ini penyusun mencoba mencari solusi permasalahan yang didapat dari tahap analisis.
- Tahap Implementasi
Merupakan tahap mengimplementasikan perancangan sistem ke situasi nyata. Pada tahap ini, mulai berurusan dengan pemeliharaan perangkat keras dan penyusunan perangkat lunak aplikasi (pengkodean).
- Tahap Pengujian
Tahap ini dapat digunakan untuk menentukan apakah sistem atau perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum.
- Tahap Pemeliharaan
Pada tahap pemeliharaan atau perawatan, mulai melakukan pengoperasian sistem dan melakukan perbaikan-perbaikan kecil (jika diperlukan)

BAB 5. HASIL YANG DICAPAI

Dalam penerapan konsep blockchain pada sistem pembuatan tanda tangan digital menggunakan QR Code, penulis melakukan beberapa tahapan hingga konsep blockchain dapat diterapkan. Konsep penerapan ini memanfaatkan basis data MySQL dimana ini merupakan penerapan konsep sederhana dalam memahami bagaimana konsep blockchain bekerja. Berikut adalah beberapa tahapan yang dilakukan.

5.2 Penambahan Tabel Baru

Kondisi saat ini tanda tangan digital berbasis QRCode disimpan pada database dengan nama tabel: **tb_qr_sign** yang memiliki atribut seperti **id_sign**, **id_user**, **id_user_mhs**, **perihal**, **tanggal**. Dalam implementasi blockchain, beberapa elemen penting yang sering ditemui adalah **blok**, **hash dari blok sebelumnya**, **timestamp**, dan **data**. Dengan mempertimbangkan struktur tabel **tb_qr_sign**, maka penulis menciptakan tabel pelengkap untuk mendukung implementasi blockchain.

- **Tabel Blok (tb_block):**

- *block_id*: ID unik untuk setiap blok.
- *previous_hash*: Hash dari blok sebelumnya. Ini membantu dalam memastikan integritas blockchain.
- *timestamp*: Waktu ketika blok dibuat.
- *hash*: Hash dari blok saat ini, yang dihitung dari kombinasi *block_id*, *previous_hash*, *timestamp*, dan *data_hash*.
- *data_hash*: Hash dari data (dalam hal ini, transaksi dari *tb_qr_sign*) yang terkait dengan blok saat ini.
- *id_dosen*

```
CREATE TABLE tb_block (  
    block_id INT AUTO_INCREMENT PRIMARY KEY,  
    previous_hash VARCHAR(255) NOT NULL,  
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP NOT NULL,  
    hash VARCHAR(255) NOT NULL,  
    data_hash VARCHAR(255) NOT NULL,  
    id_dosen  
)
```

- **Tabel Transaksi (tb_transaction):**

- *transaction_id*: ID unik untuk setiap transaksi.
- *block_id*: ID dari blok yang terkait dengan transaksi ini (foreign key ke *tb_block*).
- *id_sign*: ID tanda tangan dari *tb_qr_sign* (foreign key ke *tb_qr_sign*).
- *status*: Status transaksi, misalnya 'Pending', 'Verified', 'Rejected', dll.

```
CREATE TABLE tb_transaction (
    transaction_id INT AUTO_INCREMENT PRIMARY KEY,
    block_id INT,
    id_sign INT,
    status VARCHAR(50) NOT NULL DEFAULT 'Pending',
    FOREIGN KEY (block_id) REFERENCES tb_block(block_id),
    FOREIGN KEY (id_sign) REFERENCES tb_qr_sign(id_sign)
);
```

Setiap kali tanda tangan QR Code baru dibuat di **tb_qr_sign**, sebuah transaksi baru akan ditambahkan ke **tb_transaction**. Setiap transaksi yang ditambahkan akan memicu penciptaan blok baru di **tb_block**. Untuk setiap blok baru, Anda akan menghitung hash dari blok sebelumnya (dari **previous_hash**), timestamp saat ini, dan data transaksi terkait (dari **data_hash**) untuk menghasilkan hash blok saat ini.

Dengan pendekatan ini, penulis memastikan bahwa setiap perubahan atau manipulasi pada data akan mempengaruhi hash, yang pada gilirannya akan mengganggu integritas blockchain.

5.1 Pendefinisian Fungsi

Dalam menerapkan konsep blockchain penulis menyimpan beberapa proses didalam fungsi-fungsi yang nanti akan dipanggil oleh sistem. Adapun fungsi-fungsi tersebut adalah:

- **calculateDataHash**: Menghitung hash dari beberapa parameter yang diberikan menggunakan algoritma SHA-256 yaitu berupa data yang ditanda tangani.
- **calculateHash**: Menghitung hash dari beberapa parameter yang diberikan menggunakan algoritma SHA-256.
- **addBlock**: Menambahkan blok ke blockchain. Jika blockchain kosong, blok pertama (genesis block) akan dibuat.
- **isValid**: Memeriksa apakah blockchain yang ada saat ini valid.
- **getLatestSignData**: Mengambil data terbaru dari tabel **tb_qr_sign**.
- **getLatestBlockForUser**: Mengambil blok terakhir untuk pengguna tertentu.
- **userHasBlock**: Memeriksa apakah pengguna tertentu sudah memiliki blok di database.
- **getLastBlockHash**: Mengambil hash dari blok terakhir untuk pengguna tertentu.
- **insertBlockToDb**: Menambahkan blok ke database.
- **insertTransactionToDb**: Menambahkan transaksi ke database.
- **loadBlockchainFromDb**: Memuat blockchain dari database berdasarkan id pengguna.
- **insertDataToDb**: Menambahkan data ke tabel **tb_qr_sign**.

5.2 Pembuatan QR Code

Pembuatan QR Code diawali dengan pengguna mengisi form yang telah tersedia pada sistem, dan kemudian memilih mahasiswa yang akan diberi tanda tangan, kemudian memasukkan perihal hal yang di tanda tangani, setelah memastikan semua data terisi, maka pengguna bisa mengirimkan data, seperti pada gambar berikut.



Gambar. Form pemberian tanda tangan digital

5.3 Pembuatan Blockchain

Jika token valid pada proses sebelumnya, kode kemudian mengambil beberapa nilai dari **\$_SESSION** dan **\$_POST** dan memprosesnya. Adapun berikut penjelasan tahapan demi tahapan nya:

- **calculateDataHash:** Menghitung hash dari beberapa parameter yang diberikan menggunakan algoritma SHA-256 yaitu berupa data yang ditanda tangani.

```
function calculateDataHash($id_sign, $id_user, $id_user_mhs, $perihal, $tanggal) {  
    return hash("sha256", $id_sign . $id_user . $id_user_mhs . $perihal .  
    $tanggal);  
}
```

- **calculateHash:** Menghitung hash dari beberapa parameter yang diberikan menggunakan algoritma SHA-256.

```
function calculateHash($block_id, $previous_hash, $timestamp, $data_hash) {  
    return hash('sha256', $block_id . $previous_hash . $timestamp . $data_hash);  
}
```

- **addBlock:** Menambahkan blok ke blockchain. Jika blockchain kosong, blok pertama (genesis block) akan dibuat.

```
function addBlock($chain, $data) {
    $id_sign = $data['id_sign'];
    $id_user = $data['id_user'];
    $id_user_mhs = $data['id_user_mhs'];
    $perihal = $data['perihal'];
    $tanggal = $data['tanggal'];

    if (empty($chain)) {
        // Genesis block
        $block_id = 0;
        $previous_hash = '0'; // Hardcoded value for genesis block
        $timestamp = time();
        $hash = calculateDataHash($id_sign, $id_user, $id_user_mhs, $perihal,
$tanggal);
    } else {
        $previousBlock = end($chain);
        //$block_id = count($chain);
        $block_id = $previousBlock['block_id'] + 1;
        $previous_hash = $previousBlock['hash'];
        $timestamp = time();
        $hash = calculateDataHash($id_sign, $id_user, $id_user_mhs, $perihal,
$tanggal);
    }
    $block = [
        'block_id' => $block_id,
        'previous_hash' => $previous_hash,
        'timestamp' => $timestamp,
        'hash' => $hash,
        'data' => $data
    ];
    $chain[] = $block;
    return $chain;
}
```

- **isValid:** Memeriksa apakah blockchain yang ada saat ini valid.

```
function isValid($chain, $id_user) {
    $filteredChain = array_filter($chain, function($block) use ($id_user) {
        return $block['data']['id_user'] == $id_user;
    });
    $filteredChain = array_values($filteredChain); // Reset keys
    for ($i = 1; $i < count($filteredChain); $i++) {
        $currentBlock = $filteredChain[$i];
        $previousBlock = $filteredChain[$i - 1];
        if ($currentBlock['hash'] !== calculateDataHash(
            $currentBlock['data']['id_sign'],
            $currentBlock['data']['id_user'],
            $currentBlock['data']['id_user_mhs'],
            $currentBlock['data']['perihal'],
            $currentBlock['data']['tanggal'])) {
            return false;
        }
        if ($currentBlock['previous_hash'] !== $previousBlock['hash']) {
            return false;
        }
    }
    return true;
}
```

- **getLatestSignData:** Mengambil data terbaru dari tabel tb_qr_sign.

```
function getLatestSignData($conn) {
    $result = $conn->query("SELECT * FROM tb_qr_sign ORDER BY id_sign DESC LIMIT 1");
    if ($result && $result->num_rows > 0) {
        return $result->fetch_assoc();
    } else {
        return null;
    }
}
```

- **getLatestBlockForUser:** Mengambil blok terakhir untuk pengguna tertentu.

```
function getLatestBlockForUser($conn, $id_user) {
    $result = $conn->query("SELECT * FROM tb_block WHERE id_user = $id_user ORDER BY
    block_id DESC LIMIT 1");
    if ($result && $result->num_rows > 0) {
        return $result->fetch_assoc();
    } else {
        return null; }}
}
```

- **userHasBlock:** Memeriksa apakah pengguna tertentu sudah memiliki blok di database.

```
function userHasBlock($conn, $id_user) {
    echo "Checking for user ID: " . $id_user . "<br>";
    $result = $conn->query("SELECT 1 FROM tb_block WHERE id_user = $id_user LIMIT 1");
    if (!$result) {
        die("Query failed: " . $conn->error);
    }
    $hasBlock = $result->num_rows > 0;
    echo "userHasBlock returns: " . ($hasBlock ? "true" : "false") . "<br>";
    return $hasBlock;
}
```

- **getLastBlockHash:** Mengambil hash dari blok terakhir untuk pengguna tertentu.

```
function getLastBlockHash($conn, $id_user) {
    $result = $conn->query("SELECT hash FROM tb_block WHERE id_user = $id_user ORDER
    BY block_id DESC LIMIT 1");
    if ($result && $row = $result->fetch_assoc()) {
        return $row['hash'];
    }
}
```

- **insertBlockToDb:** Menambahkan blok ke database.

```
function insertBlockToDb($conn, $id_user, $timestamp, $hash, $data_hash) {
    if (userHasBlock($conn, $id_user)) {
        // Jika pengguna memiliki blok, ambil hash dari blok terakhir mereka.
        $previous_hash = getLastBlockHash($conn, $id_user);
    } else {
        // Jika pengguna tidak memiliki blok, ini adalah blok genesis mereka, jadi
        atur previous_hash ke '0'.
        $previous_hash = "0";
    }
    $stmt = $conn->prepare("INSERT INTO tb_block (id_user, previous_hash, timestamp,
    hash, data_hash) VALUES (?, ?, ?, ?, ?)");
    $stmt->bind_param("issss", $id_user, $previous_hash, $timestamp, $hash,
    $data_hash);
    if ($stmt->execute()) {
        return $conn->insert_id;
    } else {
        echo "Error: " . $stmt->error;
        return false;
    }
}
```

- **insertTransactionToDb:** Menambahkan transaksi ke database.

```
function insertTransactionToDb($conn, $block_id, $id_sign, $detail) {
    $stmt = $conn->prepare("INSERT INTO tb_transaction (block_id, id_sign, detail)
VALUES (?, ?, ?)");
    if ($stmt === false) {
        die("SQL Error: " . $conn->error);
    }
    $stmt->bind_param("iis", $block_id, $id_sign, $detail);
    if ($stmt->execute()) {
        return true;
    } else {
        return false;
    }
}
```

- **loadBlockchainFromDb:** Memuat blockchain dari database berdasarkan id pengguna.

```
function loadBlockchainFromDb($conn, $id_dosen) {
    $result = $conn->query("SELECT * FROM tb_block WHERE id_user = $id_dosen ORDER BY
block_id");
    $blockchain = [];
    if ($result && $result->num_rows > 0) {
        while ($row = $result->fetch_assoc()) {
            $blockchain[] = $row;
        }
    } else {
        // Jika tidak ada blok untuk id_dosen, kita buat blok genesis
        $genesisBlock = [
            'block_id' => 0,
            'previous_hash' => '0',
            'timestamp' => time(),
            'hash' => 'genesis_hash',
            'data' => 'Genesis Block for ' . $id_dosen
        ];
        $blockchain[] = $genesisBlock;
    }
    return $blockchain;
}
```

- **insertDataToDb:** Menambahkan data ke tabel tb_qr_sign.

```
function insertDataToDb($conn, $id_user, $id_user_mhs, $perihal, $tanggal) {
    $stmt = $conn->prepare("INSERT INTO tb_qr_sign (id_user, id_user_mhs, perihal,
    tanggal) VALUES (?, ?, ?, ?)");
    $stmt->bind_param("iiss", $id_user, $id_user_mhs, $perihal, $tanggal);

    if ($stmt->execute()) {
        return true;
    } else {
        return false;
    }
}
```

5.4 Contoh Hasil Pembuatan Blockchain

Berikut adalah gambaran hasil dari proses yang sudah dijabarkan sebelumnya.

```
latestSignDataArray
(
    [id_sign] => 157
    [id_user] => 1
    [id_user_mhs] => 76
    [perihal] => Percobaan 1
    [tanggal] => 2023-10-04
)
Hash value: 94124d69091b8d3767aa712493fcde7417138a97f8067d00c1e25a4980935be5
Block values:
Array
(
    [block_id] => 1
    [previous_hash] => genesis_hash
    [timestamp] => 1696392485
    [hash] => 94124d69091b8d3767aa712493fcde7417138a97f8067d00c1e25a4980935be5
    [data] => Array
        (
            [id_sign] => 157
            [id_user] => 1
            [id_user_mhs] => 76
            [perihal] => Percobaan 1
            [tanggal] => 2023-10-04
        )
)
Blockchain values:
Array
(
```

```

[0] => Array
(
    [block_id] => 0
    [previous_hash] => 0
    [timestamp] => 1696392485
    [hash] => genesis_hash
    [data] => Genesis Block for 1
)
[1] => Array
(
    [block_id] => 1
    [previous_hash] => genesis_hash
    [timestamp] => 1696392485
    [hash] => 94124d69091b8d3767aa712493fcde7417138a97f8067d00c1e25a4980935be5
    [data] => Array
        (
            [id_sign] => 157
            [id_user] => 1
            [id_user_mhs] => 76
            [perihal] => Percobaan 1
            [tanggal] => 2023-10-04
        )
    )
)

```

Blockchain is valid.

latestBlock values:

Array

```

(
    [block_id] => 0
    [previous_hash] => 0
    [timestamp] => 1696392485
    [hash] => genesis_hash
    [data] => Genesis Block for 1
)

```

newBlockId values:

1

latestBlock values:

genesis_hash

timestamp values:

2023-10-04 06:08:05

data_hash values:

94124d69091b8d3767aa712493fcde7417138a97f8067d00c1e25a4980935be5

Checking for user ID: 1

userHasBlock returns: false

block_id di database values:

1

Block and transaction data saved. Berhasil dibuat.

5.5 Memvalidasi Hasil Blockchain

Data yang biasanya divalidasi oleh akses publik adalah ``hash`` dari setiap blok karena ``hash`` ini mencerminkan integritas keseluruhan blok, termasuk ``data_hash``, ``previous_hash``, ``timestamp``, dan lainnya. Ketika memvalidasi ``hash`` dari setiap blok, secara tidak langsung juga memvalidasi ``data_hash`` karena perubahan apa pun pada ``data_hash`` akan mengubah nilai ``hash`` dari blok tersebut. Namun, jika memiliki skenario khusus di mana memungkinkan akses publik untuk memvalidasi integritas data yang sebenarnya (tanpa informasi tambahan dari blok seperti ``previous_hash``, ``timestamp``, dsb.), Anda dapat memvalidasi ``data_hash`` secara terpisah. Sehingga ada hal yang bisa di validasi, yaitu:

- Untuk memvalidasi keseluruhan integritas blok:** Gunakan ``hash``.
- Untuk hanya memvalidasi integritas data dalam blok:** Gunakan ``data_hash``.

1. Memvalidasi Hash

Untuk memvalidasi keseluruhan integritas blok.

```
<!DOCTYPE html>
<html>
<head>
    <title>Validasi Hash Blockchain</title>
</head>
<body>
    <form action="" method="post">
        <label for="hash">Masukkan Hash:</label>
        <input type="text" id="hash" name="hash" required>
        <input type="submit" value="Validasi">
    </form>
    <?php
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        include 'config.php'; // Sesuaikan dengan file koneksi database Anda
        $input_hash = $_POST['hash'];
        validateProvidedHash($input_hash);
    }
    function validateProvidedHash($hash) {
        global $conn;
        $stmt = $conn->prepare("SELECT * FROM tb_block WHERE hash = ?");
        $stmt->bind_param("s", $hash);
        $stmt->execute();
        $result = $stmt->get_result();
        if ($result && $result->num_rows > 0) {
            $block = $result->fetch_assoc();
            // Ambil data transaksi berdasarkan block_id
            $transaction = getTransactionByBlockId($block['block_id']);
            // Ambil data dari tb_qr_sign berdasarkan id_sign dari transaksi
            $qr_sign_data = getQrSignDataById($transaction['id_sign']);
```

```

// Hitung ulang hash berdasarkan data dari tb_qr_sign dan data lainnya dari blok
$recalculatedHash = calculateCompleteHash($block['block_id'],
$block['previous_hash'], $block['timestamp'], $qr_sign_data['id_sign'],
$qr_sign_data['id_user'], $qr_sign_data['id_user_mhs'], $qr_sign_data['perihal'],
$qr_sign_data['tanggal']);

    if ($recalculatedHash === $hash) {
        echo "Hash valid dan sesuai dengan data di blockchain.";
        echo "<pre>";
        print_r($block);
        echo "</pre>";
        echo "<pre>";
        print_r($qr_sign_data);
        echo "</pre>";
    } else {
        echo "Hash tidak valid dan tidak sesuai dengan data di blockchain.";
        echo "<pre>";
        print_r($block);
        echo "</pre>";
        echo "<pre>";
        print_r($qr_sign_data);
        echo "</pre>";
        echo "<pre>";
        print_r($recalculatedHash);
        echo "</pre>";
    }
} else {
    echo "Hash tidak ditemukan di blockchain.";
}
}

function getTransactionByBlockId($block_id) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM tb_transaction WHERE block_id = ?");
    $stmt->bind_param("i", $block_id);
    $stmt->execute();
    $result = $stmt->get_result();
    if ($result && $result->num_rows > 0) {
        return $result->fetch_assoc();
    } else {
        return null;
    }
}

function getQrSignDataById($id_sign) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM tb_qr_sign WHERE id_sign = ?");
    $stmt->bind_param("i", $id_sign);

```

```

$stmt->execute();
$result = $stmt->get_result();
if ($result && $result->num_rows > 0) {
    return $result->fetch_assoc();
} else {
    return null;
}
}

function calculateCompleteHash($block_id, $previous_hash, $timestamp, $id_sign,
$id_user, $id_user_mhs, $perihal, $tanggal) {
// Perlu memastikan formula hash Anda sesuai dengan yang Anda gunakan ketika
menyimpan data di blockchain.
$data_hash = hash("sha256", $id_sign . $id_user . $id_user_mhs . $perihal .
$tanggal);
    echo "<pre> block id: ";
        print_r($block_id);
        echo "</pre>";
        echo "<pre> previous_hash:";
        print_r($previous_hash);
        echo "</pre>";
    echo "<pre> Timestamp:";
        print_r($timestamp);
        echo "</pre>";

        echo "<pre> data_hash:";
        print_r($data_hash);
        echo "</pre>";

    return hash("sha256", $block_id . $previous_hash . $timestamp .
$data_hash);
?>
</body>
</html>

```

Hasil dari kode berikut adalah form seperti pada gambar berikut.

Masukkan Hash:

Gambar. Form untuk Validasi Hash.

Berikut adalah hasil validasi hash :

c0640098c5e7ed3461ecea967cbaaef91a9948d0d482b41ed62cf81b458960

Yang dapat dilihat pada gambar berikut:

Masukkan Hash:

block id: 2

previous_hash: 69f9414513dbef3735225719b73720b97e5c36ede60b030059db2d4862ae26c1

Timestamp: 2023-10-04 08:39:50

data_hash: c08e96d0be1afd6ba3241e8bbf0f52ac4f48f02517efa7676b1add90d94c91bb

Hash valid dan sesuai dengan data di blockchain.

```
Array
(
    [block_id] => 2
    [previous_hash] => 69f9414513dbef3735225719b73720b97e5c36ede60b030059db2d4862ae26c1
    [timestamp] => 2023-10-04 08:39:50
    [hash] => c0640098c5e7ed3461ecea967cbaaef91a9948d0d482b41ed62cf81b458960
    [data_hash] => c08e96d0be1afd6ba3241e8bbf0f52ac4f48f02517efa7676b1add90d94c91bb
    [id_user] => 1
)

Array
(
    [id_sign] => 158
    [id_user] => 1
    [id_user_mhs] => 142
    [perihal] => Percobaan 2
    [tanggal] => 2023-10-04
)
```

Gambar. Hasil valid dari hash yang sudah dibentuk

2. Memvalidasi Data Hash

Untuk hanya memvalidasi integritas data dalam blok.

```
<!DOCTYPE html>
<html>
<head>
    <title>Validasi Data Hash Blockchain</title>
</head>
<body>
    <form action="" method="post">
        <label for="hash">Masukkan Data Hash:</label>
        <input type="text" id="hash" name="hash" required>
        <input type="submit" value="Validasi">
    </form>
    <?php
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        include 'config.php'; // Sesuaikan dengan file koneksi database Anda
        $input_hash = $_POST['hash'];
        validateProvidedHash($input_hash);
    }
```

```

function validateProvidedHash($hash) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM tb_block WHERE data_hash = ?");
    $stmt->bind_param("s", $hash);
    $stmt->execute();
    $result = $stmt->get_result();
    if ($result && $result->num_rows > 0) {
        $block = $result->fetch_assoc();
        // Ambil data transaksi berdasarkan block_id
        $transaction = getTransactionByBlockId($block['block_id']);
        // Ambil data dari tb_qr_sign berdasarkan id_sign dari transaksi
        $qr_sign_data = getQrSignDataById($transaction['id_sign']);
        // Hitung ulang hash berdasarkan data dari tb_qr_sign
        $recalculatedHash = calculateHash($qr_sign_data['id_sign'],
        $qr_sign_data['id_user'], $qr_sign_data['id_user_mhs'], $qr_sign_data['perihal'],
        $qr_sign_data['tanggal']);
        if ($recalculatedHash === $hash) {
            echo "Hash valid dan sesuai dengan data di blockchain.";
            echo "<pre>";
            print_r($block);
            echo "</pre>";
            echo "<pre>";
            print_r($qr_sign_data);
            echo "</pre>";
        } else {
            echo "Hash tidak valid.";
        }
    } else {
        echo "Hash tidak ditemukan di blockchain.";
    }
}

function getTransactionByBlockId($block_id) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM tb_transaction WHERE block_id = ?");
    $stmt->bind_param("i", $block_id);
    $stmt->execute();
    $result = $stmt->get_result();
    if ($result && $result->num_rows > 0) {
        return $result->fetch_assoc();
    } else {
        return null;
    }
}

```

```

function getQrSignDataById($id_sign) {
    global $conn;
    $stmt = $conn->prepare("SELECT * FROM tb_qr_sign WHERE id_sign = ?");
    $stmt->bind_param("i", $id_sign);
    $stmt->execute();
    $result = $stmt->get_result();
    if ($result && $result->num_rows > 0) {
        return $result->fetch_assoc();
    } else {
        return null;
    }
}

function calculateHash($id_sign, $id_user, $id_user_mhs, $perihal, $tanggal) {
    return hash("sha256", $id_sign . $id_user . $id_user_mhs . $perihal . $tanggal);
}
?>
</body>
</html>

```

Hasil dari kode berikut adalah form seperti pada gambar berikut.

Masukkan Hash:

Gambar. Form untuk Validasi Hash.

Berikut adalah hasil validasi data hash :

c08e96d0be1afd6ba3241e8bbf0f52ac4f48f02517efa7676b1add90d94c91bb

Yang dapat dilihat pada gambar berikut:

Masukkan Data Hash:

Hash valid dan sesuai dengan data di blockchain.

```
Array
(
    [block_id] => 2
    [previous_hash] => 69f9414513dbef3735225719b73720b97e5c36ede60b030059db2d4862ae26c1
    [timestamp] => 2023-10-04 08:39:50
    [hash] => c0640098c5e7ed3461ecea967cbaaef91a9948d0d482b41ed62cf81b458960
    [data_hash] => c08e96d0be1afd6ba3241e8bbf0f52ac4f48f02517efa7676b1add90d94c91bb
    [id_user] => 1
)

Array
(
    [id_sign] => 158
    [id_user] => 1
    [id_user_mhs] => 142
    [perihal] => Percobaan 2
    [tanggal] => 2023-10-04
)
```

Gambar. Hasil valid dari data hash yang sudah dibentuk

3. Memvalidasi Hash Ketika Data Diubah

Hash dan Data Hash yang sudah di validasi pada diatas memiliki value pada atribut perihal berupa 'Percobaan 2', apa yang terjadi jika value tersebut kemudia berubah menjadi 'Percobaa 3'. Maka validasi pada Data Hash akan menampilkan hasil Tidak Valid, seperti pada gambar berikut.

Masukkan Data Hash:
Hash tidak valid.

dan hasil validasi dari Hash adalah seperti pada gambar berikut:

Masukkan Hash:

block id: 2

previous_hash: 69f9414513dbef3735225719b73720b97e5c36ede60b030059db2d4862ae26c1

Timestamp: 2023-10-04 08:39:50

data_hash: a122ba59f7a776f73ce7e6ad97882a722654a5ffe5a86f72e0e551521ed2def5

Hash tidak valid dan tidak sesuai dengan data di blockchain.

```

Array
(
    [block_id] => 2
    [previous_hash] => 69f9414513dbef3735225719b73720b97e5c36ede60b030059db2d4862ae26c1
    [timestamp] => 2023-10-04 08:39:50
    [hash] => c0640098c5e7ed3461ecea967cbaaef91a9948d0d482b41ed62cf8b81b458960
    [data_hash] => c08e96d0be1afd6ba3241e8bbf0f52ac4f48f02517efa7676b1add90d94c91bb
    [id_user] => 1
)

Array
(
    [id_sign] => 158
    [id_user] => 1
    [id_user_mhs] => 142
    [perihal] => Percobaan 3
    [tanggal] => 2023-10-04
)

8b53ebe8a7461283f1973b4f3643667ffdac3aa90621510d691ddb9a6356425f

```

5.6 Kesimpulan Hasil Blockchain

Berdasarkan percobaan yang telah dilakukan, maka terdapat beberapa kesimpulan yang bisa diambil:

- Blockchain:**
 Setiap blok mengandung **hash** dari blok sebelumnya (kecuali Genesis Block). Ini memastikan bahwa setiap blok terkait satu sama lain dalam sebuah rantai. Setiap blok memiliki **hash** uniknya sendiri yang didasarkan pada kontennya dan **hash** dari blok sebelumnya.
- Transaksi:**
Blockchain umumnya digunakan untuk mencatat transaksi. Di sini, terdapat tabel **tb_transactions** yang mencatat detail transaksi.

Berdasarkan data yang tersimpan pada database:

```
tb_block:
BID:1, PHASH:'0', HASH:'69f9414513...'. Ini adalah Genesis Block karena
previous_hash-nya adalah '0'.
BID:2, PHASH:'69f9414513...', HASH:'c0640098c5...'. Blok kedua memiliki
previous_hash yang sesuai dengan hash dari Genesis Block, yang berarti ini
mematuhi konsep blockchain.
tb_transactions:
TID:95, BID:1, QRSID:157. Transaksi pertama direferensikan ke Blok pertama dan
tb_qr_sign dengan ID 157.
TID:96, BID:2, QRSID:158. Transaksi kedua direferensikan ke Blok kedua dan
tb_qr_sign dengan ID 158.
tb_qr_sign:
QRSID:157, UID:1, UMHID:76. Detail pertama untuk QR Sign.
QRSID:158, UID:1, UMHID:142. Detail kedua untuk QR Sign.
```

Setiap blok di **tb_block** memiliki hash dari blok sebelumnya, yang sesuai dengan konsep **blockchain**. Setiap transaksi di **tb_transactions** mereferensikan sebuah blok dan sebuah entry di **tb_qr_sign**, yang menunjukkan bahwa setiap transaksi memiliki asosiasi dengan blok tertentu dan sebuah tanda tangan QR.

BAB 7. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian ini, maka dapat ditarik kesimpulan sebagai berikut; Penelitian telah berhasil mengintegrasikan teknologi blockchain ke dalam tanda tangan digital berbasis QR Code. Motivasi dari integrasi ini adalah meningkatnya kebutuhan akan solusi keamanan yang lebih tangguh dan handal di era digital saat ini. Digital signature berbasis QR Code sudah dikenal memiliki potensi yang baik dalam memastikan integritas dan keaslian informasi. Namun, dengan menggabungkannya dengan keunggulan teknologi blockchain, diharapkan akan ada peningkatan signifikan dalam hal keamanan dan keabsahan data.

Untuk meningkatkan hasil dan manfaat dari penelitian maka peneliti memberikan beberapa saran yaitu; Memanfaatkan platform blockchain lain seperti hyperledger fabric agar scalable nya bisa tercapai dengan baik.

DAFTAR PUSTAKA

Narayanan, A., et al. (2016). Bitcoin and cryptocurrency technologies. Princeton University Press.

Zheng, Z., et al. (2018). Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services.

Rivest, R. L., et al. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM.

Housley, R., et al. (2002). Internet X. 509 public key infrastructure certificate and certificate revocation list (CRL) profile. RFC 3280.

Denso Wave Incorporated. (1994). About QR Code.

Kirov, V. (2012). QR codes – application in marketing and advertising. International conference “Marketing: Experience and Perspectives”, University of Economics – Varna.

Zohar, A. (2015). Bitcoin: Under the hood. Communications of the ACM.

Tapscott, D., & Tapscott, A. (2016). Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world. Penguin.