

SKRIPSI
RANCANG BANGUN PLATFORM TANDA TANGAN
ELEKTRONIK BERBASIS BLOCKCHAIN



Disusun Oleh:
ANG SANDY KRISTIawan
17.N1.0013

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS KATOLIK SOEGIJAPRANATA
SEMARANG
2021

SKRIPSI
RANCANG BANGUN PLATFORM TANDA TANGAN
ELEKTRONIK BERBASIS BLOCKCHAIN

Diajukan dalam Rangka Memenuhi
Salah Satu Syarat Memperoleh
Gelar Sarjana Sistem Informasi



Disusun Oleh:
ANG SANDY KRISTIawan
17.N1.0013

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS KATOLIK SOEGIJAPRANATA
SEMARANG

2021

HALAMAN PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini

Nama : Ang Sandy Kristiawan

NIM : 17.N1.0013

Prodi/Konsentrasi : Sistem Informasi

Fakultas : Ilmu Komputer

Dengan ini menyatakan bahwa Laporan Skripsi dengan judul “Rancang Bangun Platform Tanda Tangan Elektronik Berbasis Blockchain” benar – benar bebas dari plagiasi, dan apabila terbukti tidak benar bersedia menerima sanksi sesuai dengan ketentuan yang berlaku.

Semarang, 21 Oktober 2021

Yang menyatakan,



Ang Sandy Kristiawan

HALAMAN PENGESAHAN



Judul Tugas Akhir: : Rancang Bangun Platform Tanda Tangan Elektronik Berbasis Blockchain
Diajukan oleh : Ang, Sandy Kristiawan
NIM : 17.N1.0013
Tanggal disetujui : 21 Oktober 2021
Telah setuju oleh
Pembimbing 1 : Prof. Dr. Ridwan Sanjaya S.E., S.Kom., MS.IEC.
Pembimbing 2 : Fx. Hendra Prasetya S.T., M.T.
Penguji 1 : Albertus Dwiyoga Widiantoro S.Kom., M.Kom.
Penguji 2 : Agus Cahyo Nugroho S.Kom., M.T.
Penguji 3 : Prof. Dr. Ridwan Sanjaya S.E., S.Kom., MS.IEC.
Ketua Program Studi : Agus Cahyo Nugroho S.Kom., M.T.
Dekan : Dr. Bernardinus Harnadi S.T., M.T.

Halaman ini merupakan halaman yang sah dan dapat diverifikasi melalui alamat di bawah ini.

sintak.unika.ac.id/skripsi/verifikasi/?id=17.N1.0013

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK
KEPENTINGAN AKADEMIS**

Sebagai civitas akademik Universitas Katolik Soegijapranata, Saya yang bertanda tangan di bawah ini :

Nama : Ang Sandy Kristiawan

Program Studi : Sistem Informasi

Fakultas : Ilmu Komputer

Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Katolik Soegijapranata Hak Bebas Royalti Non Eksklusif atas karya ilmiah saya yang berjudul : **“Rancang Bangun Platform Tanda Tangan Elektronik Berbasis Blockchain”**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Universitas Katolik Soegijapranata berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Semarang, 21 Oktober 2021

Yang menyatakan



Ang Sandy Kristiawan

KATA PENGANTAR

Segala puji syukur saya panjatkan kepada Tuhan Yang Maha Esa karena penyertaannya saya dapat menyelesaikan penulisan tugas akhir ini yang berjudul “Rancang Bangun Platform Tanda Tangan Elektronik Berbasis Blockchain”.

Di Universitas Katolik Soegijapranata Semarang skripsi menjadi salah satu syarat kelulusan untuk mendapatkan jenjang SI jurusan Ilmu Komputer program studi Sistem Informasi. Mungkin terdapat kesalahan yang saya perbuat baik sebelum maupun sesudah penulisan laporan tugas akhir skripsi ini. Untuk itu saya memohon maaf sebesar-besarnya. Dan saya bersedia menerima kritik dan saran perihal laporan yang telah saya tulis ini.

Terdapat banyak pihak yang berperan dalam hidup saya dalam pengerjaan skripsi ini di antaranya adalah:

1. Tuhan Yesus Kristus yang selalu menguatkan saya dan menyertai saya dalam situasi apapun yang saya hadapi.
2. Keluarga saya semuanya baik itu Papa, Mama, Adik saya, dan sanak keluarga saya lainnya dari keluarga besar saya.
3. UNIKA Soegijapranata yang memberikan saya berbagai pelajaran dan pengalaman yang berharga selama saya menjalani perkuliahan.
4. Dosen pembimbing saya yang senantiasa membantu saya dan selalu menyemangati saya untuk menyelesaikan penulisan ini.
5. Dan seluruh rekan-rekan seperjuangan saya yang selalu memberikan saya semangat dalam berkuliah hingga menyelesaikan laporan ini.

Akhir kata, saya mohon maaf atas segala kesalahan yang telah saya perbuat terhadap semua pihak baik yang berperan langsung dalam perkuliahan, maupun semua pihak dalam hidup saya. Saya berharap penulisan ini dapat berguna bagi siapapun yang membacanya.

Semarang, 21 Oktober 2021

Penulis



Ang Sandy Kristiawan

ABSTRAK

Seiring perkembangan teknologi, *Internet of Things* (IOT) menjadi topik yang cukup ramai diperbincangkan. Banyak kegiatan mulai dilakukan secara *online*. Salah satunya adalah kegiatan penandatanganan dokumen. PT. “XYZ” hendak membangun sebuah *prototype* aplikasi dengan *framework* yang mendukung pembuatan dan pengembangan aplikasi *multiplatform* yang dapat digunakan penggunanya untuk melakukan penandatanganan beserta penyematan sertifikat *digital*, yang kemudian bukti transaksinya dapat dicatat dan kemudian diverifikasi melalui sistem *blockchain* sebagai alat bukti validitas transaksi yang diharapkan dapat memberikan hasil yang lebih terpercaya, valid, dan aman dari berbagai serangan atau penyalahgunaan yang mungkin terjadi. Penelitian ini bertujuan untuk mengetahui bagaimana menerapkan *blockchain* dalam serangkaian aktivitas penandatanganan *digital*, bagaimana konsep dan cara melakukan verifikasi terhadap transaksi yang telah dicatat oleh sistem *blockchain*, dan membuat konsep aplikasi sederhana (*prototype*) yang menggunakan *framework* yang mudah dikembangkan apabila dibutuhkan agar aplikasi tersebut dapat dikembangkan untuk *platform* lain, tentunya juga dengan memperhatikan sistem keamanan dan tetap mudah digunakan bagi penggunanya (*client*), yang kemudian aplikasi tersebut dapat dikembangkan kembali sesuai dengan kebutuhan. Pada akhirnya, aplikasi dapat dibuat dengan mudah menggunakan *framework* Expo sebagai alat untuk mengembangkan aplikasi *client*, Node.js untuk mengembangkan aplikasi *back-end*, dan proses pencatatan transaksi serta verifikasi transaksi dapat dilakukan dengan mudah dengan adanya bantuan Hyperledger Iroha.

Kata kunci: Aplikasi, Blockchain, Framework, Internet of Things, Multiplatform, Tanda tangan digital, Sertifikat digital, Verifikasi

ABSTRACT

As technology develops, the Internet of Things (IOT) becomes a topic that is being discussed and mentioned over-and-over again. Now there are many things that can be done online. One of them is document signing activity. “XYZ” LLC wanted to build a prototype application using a framework that can build and develop a multiplatform application that can be used by the users to sign a document and attach it with a digital certificate, and the transaction activity must be recorded and can be verified through a blockchain system as a validation tools that expected to give a reliable result, valid, and secure from kind of attacks and abuse that may be happened. This study aims to find out how to apply blockchain system in a series of digital signing activities, how are the concept and how to verify a transaction that already recorded in a blockchain system, and also how to make a simple application concept (prototype) using a framework that can be easy to develop for other platforms, and of course also paying attention to the security system and make the application remains easy to use for its users (clients), which then the application can be redeveloped according to needs. In the end, the application can be made easily using Expo framework as a tool to develop the client application, Node.js to develop the back-end application, and the process for recording and verifying transactions can be done easily with the help of Hyperledger Iroha.

Kata kunci: Application, Blockchain, Digital Certificate, Digital Signature, Framework, Internet of Things, Multiplatform, Verification

DAFTAR ISI

HALAMAN COVER	1
HALAMAN JUDUL	2
HALAMAN PERNYATAAN KEASLIAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	v
KATA PENGANTAR	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xii
DAFTAR KODE	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	3
BAB II TINJAUAN PUSTAKA	4
2.1 Tanda Tangan Elektronik.....	4
2.2 Public Key Infrastructure	4
2.3 Sertifikat Digital (Sertifikat elektronik).....	4
2.4 Kriptografi.....	5
2.5 Blockchain	5

2.6 Node.js	5
2.7 React	6
2.8 Expo	6
2.9 MongoDB	7
2.10 JWT (JSON Web Token).....	8
2.11 OpenXPKI.....	10
2.12 Hyperledger Iroha	10
2.13 Docker.....	15
2.14 Oracle VirtualBox	15
BAB III METODOLOGI PENELITIAN	17
3.1 Kerangka Pikir	17
3.2 Persiapan Pra-pengembangan	17
3.2 Pengembangan Aplikasi.....	18
3.3 Pengujian Keamanan Aplikasi.....	19
3.3.1 Pengujian terhadap konsistensi data Hyperledger Iroha.....	19
3.3.2 Pengujian terhadap dokumen yang telah melewati proses sertifikasi.....	19
3.4 Flowchart Penelitian	19
3.5 Konsep Aplikasi.....	21
3.6 Alasan Penggunaan Teknologi	21
3.6.1 Sistem REST API dan Front-end berbasis Javascript.....	21
3.6.2 MongoDB sebagai database	22
3.6.3 OpenxPKI	23
3.6.4 Hyperledger Iroha	23
BAB IV PERANCANGAN DAN PENGUJIAN APLIKASI.....	25
4.1 Alur aplikasi.....	25
4.1.1 Pendaftaran (Signing up)	25
4.1.2 Masuk ke aplikasi (Signing in)	48
4.1.3 Penggunaan layanan penandatanganan secara individual (Individual Signing).....	58

4.1.4 Penggunaan layanan penandatanganan secara kelompok (multiple signing)	82
4.1.5 Penggunaan layanan enterprise (group) signing pada sisi enterprise admin	89
4.1.6 Penggunaan layanan enterprise (group) signing pada sisi enterprise member	91
4.1.7 Verifikasi Transaksi Penandatanganan	101
4.1.8 Signing Out (Log-out).....	111
4.4 Pengujian keamanan Hyperledger Iroha	115
4.4.1 Simulasi apabila sebuah peer kehilangan block dengan posisi urutan block yang tidak berurutan	115
4.4.2 Simulasi apabila salah satu block dalam direktori “blockstore” sebuah peer diubah	116
4.4.3 Simulasi apabila terdapat beberapa peer yang mempunyai kondisi direktori “blockstore” yang berbeda.....	116
4.4.4 Simulasi apabila semua data pada “blockstore” peer dalam jaringan blockchain disamakan kondisinya.....	117
4.4.5 Simulasi perubahan data pada “World State View”	117
4.5 Temuan permasalahan pada Hyperledger Iroha	119
4.6 Pengujian keamanan dokumen yang telah disertifikasi	124
BAB V KESIMPULAN DAN SARAN	129
5.1 Kesimpulan	129
5.2 Saran	130
DAFTAR PUSTAKA	131
LAMPIRAN.....	133
A. Perancangan User Interface	133
B. Persiapan Pengembangan Sistem (Backend).....	134
B.1 Node.Js	134
B.2 MongoDB	146
B.3 Hyperledger Iroha	152
B.4 OpenxPKI.....	167
C. Persiapan Pengembangan Aplikasi Front-end.....	170

DAFTAR GAMBAR

Gambar 2.1 Konversi JSON menjadi bentuk BSON.....	7
Gambar 2.2 Perbedaan JSON dan BSON.....	8
Gambar 2.3 SAML VS JWT Format.....	9
Gambar 2.4 Sistem pada Hyperledger Iroha.....	11
Gambar 2.5 Struktur blockchain dan struktur data block pada Hyperledger Iroha	13
Gambar 2.6 Mengakses konten blockstore melalui terminal.....	14
Gambar 2.7 Mengakses konten WSV menggunakan DBeaver	14
Gambar 2.8 Docker container Hyperledger Iroha.....	15
Gambar 3.1 Kerangka Pikir.....	17
Gambar 3.2 Alur pengembangan aplikasi.....	18
Gambar 3.3 Flowchart Penelitian.....	20
Gambar 4.1 Halaman awal saat aplikasi dibuka, dan halaman pilihan tipe akun yang hendak didaftarkan.....	25
Gambar 4.2 Modal berbentuk drawer untuk memilih new account atau join existing group..	26
Gambar 4.3 QR Code untuk mendaftarkan anggota enterprise group.....	26
Gambar 4.4 Calon group member melakukan scan QR Code dari group admin	27
Gambar 4.5 Halaman persetujuan layanan.....	27
Gambar 4.6 Halaman pendaftaran akun tipe individual.....	28
Gambar 4.7 Field tambahan pada pendaftaran enterprise admin account.....	29
Gambar 4.8 Field yang dapat diisi oleh enterprise member.....	29
Gambar 4.9 Pengambilan foto kartu identitas dan selfie bersama kartu identitas.....	30
Gambar 4.10 Lanjutan halaman pendaftaran pengguna individual.....	30
Gambar 4.11 Menerima Kode OTP melalui SMS.....	35
Gambar 4.12 Kode OTP yang dimasukkan tidak sesuai.....	35
Gambar 4.13 Tampilan apabila pendaftaran berhasil.....	42

Gambar 4.14 Respon dari OpenXPKI apabila request belum di-approve RA Admin.	42
Gambar 4.15 Group admin menyetujui request dan menunggu persetujuan RA operator.	43
Gambar 4.16 Web dashboard Request Authority yang dibuat menggunakan Expo.	44
Gambar 4.17 Menu “Need Approval” pada dashboard apabila menerima request pendaftaran.	44
Gambar 4.18 Apabila request yang masuk ke OpenXPKI belum disetujui.	45
Gambar 4.19 Tampilan web OpenXPKI ketika terdapat request.	45
Gambar 4.20 Tampilan detail request dan juga pilihan untuk menerima atau menolak request.	45
Gambar 4.21 Tampilan apabila request enrollment disetujui.	46
Gambar 4.22 Tampilan apabila melakukan persetujuan melalui dashboard admin operator RA.	46
Gambar 4.23 Tidak ada request yang masuk atau semua request telah ditangani oleh RA operator.	47
Gambar 4.24 Tampilan pada daftar member group apabila member telah disetujui oleh RA operator.	47
Gambar 4.25 Certificate chain yang diberikan oleh OpenXPKI.	48
Gambar 4.26 Tampilan halaman “Sign In”	49
Gambar 4.27 Tampilan apabila username atau password salah.	50
Gambar 4.28: Pilihan menu Sign and Certify PDF pengguna non-enterprise.	58
Gambar 4.29 Menu Sign Document.	59
Gambar 4.30 Proses meletakkan letak container tanda tangan.	59
Gambar 4.31 Melakukan tanda tangan dengan tanda tangan baru.	60
Gambar 4.32 Tampilan tanda tangan yang telah tersimpan.	60
Gambar 4.33 Halaman konfirmasi dan tampilan apabila dokumen ditemukan pada sistem. ..	70
Gambar 4.34 Pop-up apabila transaksi telah selesai dilakukan oleh penandatanganan terakhir, dan tampilan dokumen yang telah dibuka.	81

Gambar 4.35 Dokumen selesai melewati proses penandatanganan.....	82
Gambar 4.36 Halaman konfirmasi pada saat pembuatan “Signing Request” (proposal).	92
Gambar 4.37 Halaman konfirmasi pembuatan proposal setelah semua data siap dikirimkan.	93
Gambar 4.38 Pop-up (modal) setelah proposal berhasil terkirim.\.....	95
Gambar 4.39 Proposal pada halaman “Signing Request”.....	95
Gambar 4.40 Tampilan proposal pada akun admin.	96
Gambar 4.41 Admin enterprise melakukan konfirmasi terhadap proposal.	97
Gambar 4.42 Proposal telah disetujui oleh admin enterprise.	98
Gambar 4.43 Pengguna yang diminta untuk menandatangani menerima request.	99
Gambar 4.44 Menandatangani dokumen.	99
Gambar 4.45 Tampilan halaman konfirmasi pada penandatanganan “non-initiator”.	100
Gambar 4.46 Hasil dari proses penandatanganan.....	100
Gambar 4.47 Detail dokumen di menu “My Signing Request” pada tab “Finished”.....	101
Gambar 4.48 Mengakses menu “Verify PDF” pada aplikasi client.....	102
Gambar 4.49 Verifikasi manual melalui file.....	107
Gambar 4.50 Detail verifikasi dokumen melalui upload.....	108
Gambar 4.51 Menyimpan dokumen yang telah melewati proses penandatanganan.	108
Gambar 4.52 Verifikasi dokumen asli “sebelum ditandatangani”.....	109
Gambar 4.53 Hasil dokumen yang ditandatangani melalui platform lain, dan tanpa tanda tangan.....	110
Gambar 4.54 Verifikasi transaksi penandatanganan melalui scan QR Code.....	111
Gambar 4.55 Detail transaksi penandatanganan melalui scan QR Code.....	111
Gambar 4.56 Melakukan sign out pada aplikasi client.	114
Gambar 4.57 Tampilan log ketika error karena peer tidak berurutan.....	115
Gambar 4.58 Tampilan log saat melakukan sinkronisasi blockstore.....	115
Gambar 4.59 Kondisi blockstore saat pertama kali dijalankan dan sedang dalam proses sinkronisasi.	116

Gambar 4.60 Mengganti data detail akun pada WSV salah satu peer.	117
Gambar 4.61 Hasil query “getAccount” ke peer yang bermasalah (data WSV diubah secara paksa).	118
Gambar 4.62 Hasil query “getAccount” ke peer yang data WSV tidak diubah secara paksa.	118
Gambar 4.63 Bug yang mengakibatkan peer harus dimulai ulang.	119
Gambar 4.64 Contoh payload yang diberikan pada API query “getTransactions”.	121
Gambar 4.65 Lanjutan contoh payload yang masih berlanjut.	121
Gambar 4.66 Console-logging payload transaksi dan mengambil created time pada payload.	122
Gambar 4.67 Contoh format informasi balikan query “getAccountAssetTransactions”.....	123
Gambar 4.68 Dokumen telah selesai melewati proses transaksi.	124
Gambar 4.69 Detail dari salah satu tanda tangan apabila diklik.	125
Gambar 4.70 Detail sertifikat penandatanganan pertama.	126
Gambar 4.71 Detail dari intermediate certificate.	126
Gambar 4.72 Detail sertifikat dari penandatanganan kedua.	127
Gambar 4.73 Annotation tools menjadi tidak berfungsi karena permission.	127
Gambar 4.74 Perbandingan dengan PDF yang tidak diamankan dengan permission sama sekali.	127
Gambar 4.75 Sertifikat yang menjadi invalid karena perubahan pada dokumen.	127
Gambar 4.76 Hasil apabila dokumen tidak dikunci dengan permission.	128
Gambar A.1: Desain kasar melalui Figma.	133
Gambar A.2: Preview design Figma secara interaktif.	134
Gambar B.1: Tampilan nvm pada terminal.	135
Gambar B.2: Hasil perintah “nvm ls-remote”	135
Gambar B.3: Mengunduh dan menginstall Node versi lts/fermium	136
Gambar B.4: Contoh struktur folder project aplikasi backend.	140

Gambar B.5: Tampilan mengakses database melalui MongoDB shell	149
Gambar B.6: Mengakses database melalui MongoDB Compass	149
Gambar B.7: Contoh tampilan data pada collections.....	150
Gambar B.8: Contoh operasi yang dapat dilakukan dengan model yang telah dibuat.	152
Gambar B.9: Mengambil data pada model yang telah dibuat menggunakan package “mongoose”.....	152
Gambar B.10: Versi image Hyperledger Iroha yang hendak digunakan.	153
Gambar B.11: Commands yang dituliskan pada file “genesis.block”.....	154
Gambar B.12: Lanjutan commands yang dituliskan pada file “./genesis.block”.....	155
Gambar B.13: Script “entrypoint-genesis.sh” dalam pembuatan keypair beserta genesis block.	156
Gambar B.14: Pengkopian file key yang telah dibuat melalui container iroha_genesis.	156
Gambar B.15: Hasil dari key yang telah berhasil di copy ke directory local.	157
Gambar B.16: Membuat container yang diperlukan melalui file “docker-compose.yml”. ...	157
Gambar B.17: Masuk ke directory salah satu peer dan melihat konten dari blockstore.....	158
Gambar B.18: Masuk ke iroha-cli dan juga melakukan query menggunakan iroha-cli.	159
Gambar B.19: Hyperledger Iroha Explorer yang dikembangkan oleh diva.exchange.	160
Gambar B.20: Hasil percobaan sukses memanggil endpoint “./getaccount”.....	167
Gambar B.21: Menjalankan script konfigurasi sampel pada terminal.....	168
Gambar B.22: Tampilan log bila OpenXPKI sukses terinstall dengan script sampel konfigurasi.	168
Gambar B.23: Tampilan respon sukses endpoint “/rpc/enroll/openapi-spec”.....	169
Gambar B.24: Website yang dihasilkan oleh script konfigurasi sampel diakses melalui web browser.....	169
Gambar C.1: Contoh struktur folder React Component.	173

DAFTAR KODE

Kode 4.1 React Component “OtpModal” yang akan muncul apabila state “otpModal” bernilai true.	31
Kode 4.2 React Component “OtpModal”.	33
Kode 4.3 Kode pada endpoint “./sendotp” pada backend mengirim OTP menggunakan vendor Twilio.....	34
Kode 4.4 Pembuatan CSR untuk dikirimkan ke OpenXPKI dan data CSR yang akan disimpan.....	37
Kode 4.5 Detail fungsi “csrPkcs10”.	38
Kode 4.6 Fungsi yang digunakan untuk melakukan berbagai POST request menggunakan package “node-fetch”.....	39
Kode 4.7 Fungsi yang digunakan untuk melakukan request certificate ke OpenXPKI.....	40
Kode 4.8 Melakukan request certificate dan menyimpan ke dalam database.....	41
Kode 4.9 Source code pada endpoint “./signin”.....	51
Kode 4.10 Kumpulan fungsi dalam proses pembuatan, penginputan, dan penghapusan token juga refresh token.....	53
Kode 4.11 Fungsi “signInAsync” pada aplikasi frontend.....	54
Kode 4.12 Fungsi “verifyOtp” dan “onOtpDone”.....	55
Kode 4.13 Fungsi “storeToken” yang akan dipanggil apabila proses verifikasi OTP berhasil.	55
Kode 4.14 Pengecekan apakah terdapat token tersimpan pada perangkat saat aplikasi dijalankan.....	56
Kode 4.15 Middleware “authentication”.	57
Kode 4.16 Contoh menggunakan middleware “authentication” pada sebuah route.....	58
Kode 4.17 Menjalankan sekali engine PDFTron pada aplikasi client saat halaman dimuat. ..	61
Kode 4.18 Fungsi “retrieveSignatures”.	62
Kode 4.19 Fungsi untuk memunculkan browser file pada ponsel dengan filter format file PDF.	63

Kode 4.20 React Component pada halaman “PlaceSignatureScreen”.....	64
Kode 4.21 Fungsi “onFormFieldValueChanged” dan “onAnnotationChanged”	66
Kode 4.22 Konfigurasi toolbar apa saja yang boleh ditampilkan PDFTron.....	66
Kode 4.23 Fungsi “correction”.....	66
Kode 4.24 Fungsi untuk melakukan pengecekan apabila tombol “Next” ditekan.	67
Kode 4.25 Fungsi “attachLocation”.....	68
Kode 4.26 Validasi terhadap data yang akan dikirimkan.	68
Kode 4.27 Contoh sederhana format fungsi pada aplikasi client untuk mengirimkan data tanda tangan.	70
Kode 4.28 Pengecekan ada tidaknya tanda tangan serta apakah dokumen ditemukan pada database.....	71
Kode 4.29 Fungsi “CheckHasSignature”.....	71
Kode 4.30 Pengecekan apakah pengguna mempunyai jumlah saldo mencukupi.....	72
Kode 4.31 Fungsi untuk melakukan query “getAccountAssets” ke Hyperledger Iroha.....	73
Kode 4.32 Fungsi yang digunakan untuk melakukan hashing dokumen.....	73
Kode 4.33 Fungsi untuk membuat arsip PKCS#12.....	74
Kode 4.34 Pembuatan dan konversi QRCode.....	74
Kode 4.35 Fungsi yang membantu dalam proses pengkonversian string base64 ke bentuk buffer.....	74
Kode 4.36 Menjalankan fungsi “CertifyPDF” kemudian menangkap responnya.	75
Kode 4.37 Fungsi “CertifyPDF”.....	77
Kode 4.38 Melakukan proses pencatatan transaksi dalam system blockchain.....	78
Kode 4.39 Menyimpan tanda tangan yang disertakan dalam request.....	78
Kode 4.40 Fungsi “saveUserSignature” yang digunakan untuk menyimpan tanda tangan.....	79
Kode 4.41 Fungsi untuk melakukan input ke database setelah semua aktivitas dilakukan, serta respon API yang memberikan dokumen yang telah disertifikasi.	80
Kode 4.42 Fungsi “runWithCleanup”.....	80

Kode 4.43 Membuka file yang telah tersimpan pada aplikasi client.	81
Kode 4.44 Proses validasi dokumen yang dikirimkan.....	84
Kode 4.45 Proses validasi dokumen terhadap database dan tindakan selanjutnya.....	84
Kode 4.46 Pengecekan lebih lanjut terhadap dokumen yang ditemukan pada database.	85
Kode 4.47 Logika untuk menentukan tindakan penandatanganan tahap terakhir.	86
Kode 4.48 Fungsi sertifikasi dokumen pada tahap terakhir.....	86
Kode 4.49 Fungsi untuk mencatat perubahan ke database.	86
Kode 4.50 Logika untuk melakukan proses sertifikasi tanpa mengunci dokumen.....	87
Kode 4.51 Fungsi untuk mencatat perubahan ke database dengan status on process.....	87
Kode 4.52 Proses menyimpan data request penandatanganan.....	89
Kode 4.53 Mengecek apakah admin mempunyai saldo cukup untuk melakukan transaksi enterprise.....	90
Kode 4.54 Perpindahan saldo dari admin group ke membernya.	91
Kode 4.55 Objek body request pada fungsi “transferAdminAssetToMember”	91
Kode 4.56 Validasi terhadap dokumen proposal yang dikirimkan.....	94
Kode 4.57 Fungsi apabila dokumen tidak ditemukan pada sistem.....	94
Kode 4.58 Melakukan perubahan status berdasarkan keputusan admin enterprise.....	97
Kode 4.59 Proses verifikasi dengan dua metode.	104
Kode 4.60 Fungsi yang menangani query “getTransactions” ke Hyperledger Iroha.	105
Kode 4.61 Melakukan verifikasi transaksi dengan query “getAccountAssetTransactions”..	106
Kode 4.62 Query “getAccountTransactions” ke Hyperledger Iroha.	107
Kode 4.63 Fungsi yang digunakan saat melakukan sign out pada aplikasi client.	112
Kode 4.64 Proses pada endpoint “./signout” pada aplikasi backend.	113
Kode 4.65 Melakukan pembuatan ulang container Hyperledger Iroha.	119
Kode B.1 Contoh kode pada protected route.	141
Kode B.2 Middleware “authentication”.....	143

Kode B.3 REST API menggunakan express.....	144
Kode B.4 Middleware error handler.	146
Kode B.5 Melakukan instalasi MongoDB pada sistem Ubuntu menggunakan package tarball (tgz).	147
Kode B.6 Menginstall MongoDB Compass (GUI MongoDB).....	147
Kode B.7 Command untuk membuat root user pada MongoDB.	147
Kode B.8 Mengaktifkan MongoDB dengan fitur autentikasi pada port yang dipilih.	148
Kode B.9 Masuk ke MongoDB CLI sekaligus melakukan autentikasi pengguna.	148
Kode B.10 Command pada terminal untuk melakukan autentikasi di dalam MongoDB CLI.	148
Kode B.11 Contoh membuat model atau schema collection “users”	151
Kode B.12 Membuat koneksi ke MongoDB menggunakan package “mongoose”	151
Kode B.13 Command untuk menjalankan script pembuatan genesis block serta pembuatan key.....	157
Kode B.14 Command untuk menjalankan container pada docker.....	158
Kode B.15 Command untuk masuk ke dalam direktori container peer Hyperledger Iroha. .	158
Kode B.16 Command service dan query service.	162
Kode B.17 Melakukan pembuatan akun baru pada sistem blockchain Hyperledger Iroha. ..	162
Kode B.18 Command options dan query options.	163
Kode B.19 Contoh melakukan query dari sebuah akun.....	163
Kode B.20 Contoh melakukan command “createAccount” dengan pendekatan “objek transaksi”.....	164
Kode: B.21 Fungsi untuk membuat hash dari suatu transaksi Hyperledger Iroha.....	164
Kode B.22 Membuat transaksi batch.	165
Kode B.23 Contoh mendapatkan hash dari masing-masing transaksi pada batch.....	166
Kode B.24 Command untuk menginstall expo-cli dan react-native-cli secara global.....	170

Kode B.25 Mendaftarkan kode copy-webviewer-files yang akan berjalan begitu project dijalankan.....171

Kode B.26 File “package.json” yang berada pada direktori React component PDFTron.....173

