

## Rancang Bangun Aplikasi Donasi Terdesentralisasi Berbasis Blockchain

Alfrio Matthew<sup>1</sup>, Muhammad Anno Suwarno<sup>2</sup>  
Universitas Persada Indonesia Y.A.I<sup>1,2</sup>  
E-mail: alfriomattew@gmail.com<sup>1</sup>, suwarno@upi-yai.ac.id<sup>2</sup>

### ABSTRAK

Donasi adalah suatu kegiatan dimana seseorang ataupun sekelompok orang menyumbangkan sesuatu kepada individu atau kelompok dalam bentuk uang atau barang. Di zaman sekarang ini, hampir semuanya dapat diakses secara daring, salah satunya adalah kegiatan berdonasi. Seiring dengan tingginya kesadaran masyarakat dalam berdonasi dan besarnya dana yang dikelola, banyak pula permasalahan yang ada dalam sistem donasi, seperti penggelapan dana maupun ketidakterbukaan dalam hal pendistribusian. Untuk mengatasi hal tersebut, maka dibutuhkan suatu sistem yang dapat memberikan informasi transaksi secara transparan. Tujuan dari penelitian ini adalah membuat sebuah aplikasi donasi berbasis teknologi blockchain karena dengan menggunakan sistem berbasis blockchain, maka proses donasi dapat dilakukan dengan lebih transparan. Dengan menggunakan smart contract yang berjalan pada jaringan blockchain, transaksi yang dilakukan pada aplikasi donasi ini tersimpan di dalam basis data publik sehingga lebih terbuka dan tidak dapat diubah oleh siapapun.

**Kata kunci :** *donasi, blockchain, aplikasi terdesentralisasi*

### ABSTRACT

Donation is an activity where a person or group of people donate something to an individual or group in the form of money or goods. This day, almost everything can be accessed online, and donation activities is one of them. Along with the high awareness of the public in donating and the big amount of funds managed, there are also many problems in the donation system, such as embezzlement of funds and non-disclosure in terms of distribution. To overcome this, we need a system that is able to provide transaction information transparently. The purpose of this research is to create a donation application based on blockchain technology because by using a blockchain-based system, the donation process can be done more transparently. By using smart contracts that run on a blockchain network, transactions made in this donation application are stored in the public database so it is more open and cannot be changed by anyone.

**Keyword :** *donation, blockchain, decentralized application*

## 1. PENDAHULUAN

Membantu sesama merupakan salah satu kewajiban bagi setiap umat manusia. Banyak di antara masyarakat mempercayakan bantuannya melalui platform penyaluran bantuan sosial. Pada era digital sekarang ini, penggalangan

dana juga mengalami kemajuan di platform digital yang disebut *crowdfunding*. Tren donasi melalui platform digital justru mengalami peningkatan secara signifikan. Donatur yang sebagian besar berasal dari generasi milenial, menjadi salah satu donatur

potensial yang menyalurkan sumbangan melalui platform digital.

Penelitian ini memfokuskan pada implementasi teknologi yang dapat membantu transparansi donasi untuk mengurangi kecurangan yang selama ini dilakukan beberapa oknum. Untuk itu diperlukan suatu metode baru untuk mengatasi kelemahan dari sistem konvensional ini. Blockchain merupakan mesin transparansi di mana siapa pun dapat bergabung dengan jaringan dan dapat melihat semua informasi yang direkam di dalamnya. Blockchain dengan metode *read and write* merupakan salah satu solusi untuk mengatasi masalah transparansi tersebut.

Cara kerja blockchain didasari pada struktur di dalamnya, setiap catatan yang disebut *block* akan dihubungkan satu sama lain dalam satu daftar panjang yang dikenal dengan *chain*. Terdapat 3 (tiga) sifat utama blockchain yang membuat teknologi ini memiliki pengaruh yang kuat. Sistem desentralisasi blockchain adalah sistem yang tidak memiliki otoritas sentral. Desentralisasi juga dapat diartikan sebagai suatu sistem yang keseluruhan pengambil keputusan diserahkan kepada pengguna sistem. Blockchain bersifat transparan dimana seluruh transaksi yang ada di dalamnya dapat diakses oleh publik, di saat yang sama penggunaan blockchain juga menjamin privasi penggunaannya. *Immutability* atau kekekalan dalam konteks blockchain adalah semua jejak digital transaksi yang telah terinput dalam blockchain tidak dapat dirusak atau diubah. Cara kerja tersebut sangat penting bagi lembaga keuangan. Diharapkan dengan adanya penerapan sistem blockchain dapat menjadi jawaban dari permasalahan yang ada.

## 2. LANDASAN TEORI

### 2.1 Blockchain

Blockchain merupakan sebuah buku besar yang terdistribusi dimana

blockchain terdiri dari blok (*block*) yang saling berhubungan dan membentuk pola rantai (*chain*). Seluruh transaksi yang terjadi pada jaringan blockchain akan tersimpan ke dalam daftar yang disebut dengan blok. Setiap blok memiliki *block number*, *timestamp*, *transaction data*, dan *cryptography hash* dari *previous block* sehingga data yang ada berkesinambungan dan membentuk pola rantai seperti konsep dari blockchain (Prashanth 2018).

Pada pengaplikasiannya, blockchain dapat digunakan untuk menulis suatu transaksi yang memuat nilai *hash* dari suatu dokumen digital yang divalidasi oleh seluruh node yang berada di dalamnya. Blockchain memiliki sifat tidak bisa diubah yang berguna untuk mencegah adanya manipulasi dan *trust issue* akan adanya perbedaan hash dari suatu dokumen digital. Selain itu, semua transaksi yang sudah ditulis pada blockchain akan selamanya tersimpan pada blockchain tersebut dan tidak akan pernah bisa dihapus.

Menurut pendapat Yaga (Yaga 2019), blockchain adalah *ledger* atau buku besar digital yang peredarannya dari transaksi serta ditandatangani dengan kriptografi dan dibagi-bagi ke dalam blok. Setiap blok akan dihubungkan secara kriptografi pada hash blok sebelumnya setelah dilakukan validasi serta atas keputusan konsensus. Namun, prinsip blockchain dan Distributed Ledger Technology (DLT) merupakan dua teknologi berbeda yang dianggap teknologi tersebut sama atau serupa karena banyaknya pemahaman kurang tepat (*misconception*). Hubungan antara *distributed ledger* dan blockchain adalah blockchain merupakan hasil dari pengembangan Distributed Ledger Technology (DLT). Pada tahun 1991 merupakan rekam jejak awal konsep blockchain oleh peneliti bernama Stuart Haber dan W. Scott Stornetter yang menulis sebuah paper dengan judul "How to Time-Stamp a Digital Document". Dari

paper tersebut membahas tentang implementasi untuk melakukan verifikasi dokumen saat sedang dibuat atau diubah. Kemudian teknologi blockchain ini mengalami peningkatan signifikan pada tahun 2005 oleh ilmuwan komputer bernama Nick Szabo yang memberikan konsep lengkap mengenai Blockchain-Based Currency bernama bit.gold.

## 2.2 Smart Contract dan Solidity

Menurut pendapat Luu (Luu 2016), *smart contract* adalah program yang berjalan pada blockchain dan memiliki eksekusi yang benar yang ditegakkan oleh protokol konsensus. Dalam sebuah *smart contract* terdapat kode yang menjadi aturan yang mana akan dieksekusi oleh sebuah event, dalam hal ini transaksi seperti pembayaran, pembacaan data, dan penulisan data.

*Smart contract* bersifat publik pada Ethereum Blockchain sehingga dapat diakses oleh *smart contract* sampai batasan tertentu sesuai aturan yang diterapkan pada *smart contract* yang akan diakses.

*Solidity* merupakan bahasa pemrograman tingkat tinggi yang diperuntukkan untuk menulis kontrak yang ditargetkan pada Ethereum *virtual machine*. *Solidity* digunakan untuk menulis kode dari *smart contract* yang nantinya akan mengeksekusi transaksi pada blockchain.

## 2.3 Aplikasi Terdesentralisasi

Dalam dokumentasi Ethereum, aplikasi terdesentralisasi (DApp) adalah aplikasi yang dibangun di atas jaringan terdesentralisasi yang menggabungkan *smart contract* dan antarmuka web. Di Ethereum, *smart contract* dapat diakses dan transparan seperti API terbuka sehingga DApp dapat menyertakan *smart contract* yang telah ditulis orang lain. Secara sederhana sebuah aplikasi terdesentralisasi merupakan aplikasi yang mengintegrasikan *tech stack* dengan jaringan blockchain. Hal ini termasuk

integrasi *user* dengan jaringan blockchain, blockchain dengan jaringan peer-to-peer dan lain sebagainya. Aplikasi terdesentralisasi dapat berjalan secara server-less atau tanpa server karena semua yang terjadi di dalamnya hanyalah operasi *read and write* dari blockchain.

Aplikasi terdesentralisasi adalah bagian dari perangkat lunak yang berkomunikasi dengan blockchain, yang mengelola status semua aktor jaringan. Antarmuka aplikasi terdesentralisasi tidak terlihat berbeda dari situs web atau aplikasi web. *Smart contract* mewakili logika inti dari aplikasi yang terdesentralisasi. *Smart contract* adalah blok bangunan integral dari blockchain, yang memproses informasi dari sensor atau peristiwa eksternal dan membantu blockchain mengelola keadaan semua aktor jaringan.

Aplikasi terdesentralisasi mirip dengan aplikasi web tradisional. Frontend pada aplikasi menggunakan teknologi yang sama dengan aplikasi web untuk merender halaman. Aplikasi tersebut berisi "dompet" yang berkomunikasi dengan blockchain. Dompet mengelola kunci kriptografi dan alamat blockchain. Infrastruktur kunci publik digunakan untuk identifikasi dan otentikasi pengguna. Dalam aktivitas ini, bukan API yang terhubung ke database, melainkan dompet yang menjadi pemicu aktivitas *smart contract* yang berinteraksi dengan blockchain (Voshmgir 2019).

## 2.4 Cryptocurrency Wallet dan Metamask

*Cryptocurrency wallet* merupakan aplikasi yang digunakan untuk menyimpan akun pengguna pada jaringan blockchain. Pada umumnya *cryptocurrency wallet* menyimpan sebuah *private key* dari akun yang terdaftar pada jaringan blockchain. *Cryptocurrency wallet* dijadikan sebagai media antara pengguna dengan jaringan blockchain. *Cryptocurrency wallet* dapat digunakan untuk melakukan penandatanganan dari

sebuah transaksi yang dibuat oleh pengguna.

Metamask merupakan salah satu teknologi *cryptocurrency wallet* berbentuk aplikasi yang mudah digunakan karena berupa *extension* pada web *browser*. Hal ini menjadikan metamask sangat cocok untuk pengembangan DApp berbasis web. Beberapa *cryptocurrency wallet* yang sering digunakan adalah Metamask, Trust Wallet, dan MyEtherWallet.

### 2.5 Web3.js

Web3.js merupakan sekumpulan library javascript yang dikembangkan untuk berinteraksi dengan node Ethereum dengan protokol HTTP, IPC, atau web socket. Web3.js dapat digunakan sebagai media yang digunakan DApp untuk berinteraksi dengan *smart contract* pada jaringan Ethereum. Penggunaan web3.js mempermudah pengembangan aplikasi yang memerlukan interaksi dengan Ethereum.

### 2.6 Ethereum

Ethereum merupakan teknologi yang berdasar pada blockchain sama seperti Bitcoin. Ethereum sendiri pertama kali dikenalkan oleh Vitalik Buterin pada 2013. Ethereum ditujukan sebagai protokol alternatif untuk membangun aplikasi yang terdesentralisasi. Ethereum menyediakan sebuah fungsionalitas *smart contract* untuk mendukung hal tersebut. Ethereum juga mendukung eksekusi *script* yang ditulis pada bahasa pemrograman berlevel tinggi dengan menyediakan sebuah virtual machine yang dikenal sebagai Ethereum Virtual Machine.

### 2.7 React Native

Menurut Prashanth (Prashanth 2018), React adalah open-source library JavaScript deklaratif, efisien dan fleksibel untuk membangun antarmuka pengguna. React memungkinkan untuk membuat *user interface* yang kompleks dengan set

kode kecil yang terisolasi yang disebut "komponen". React JS ini digunakan untuk menangani lapisan tampilan dalam aplikasi satu halaman dan pengembangan *mobile application*.

## 3. METODOLOGI

### 3.1 Analisis

Pada tahap analisis ini bertujuan untuk menemukan permasalahan dalam donasi. Setelah melakukan pengumpulan data dengan metode observasi dan studi literatur, ditemukan beberapa permasalahan yang menjadi motivasi untuk membuat sebuah aplikasi web untuk donasi dengan memanfaatkan teknologi blockchain sehingga aplikasi yang dibangun dapat mengatasi permasalahan tersebut, diantaranya adalah pada aplikasi ini pembukuan donasi bersifat transparan, tidak bisa diubah, dan bersifat permanen sehingga transparansi dapat terjaga dengan baik.

#### 3.1.1 Identifikasi Masalah

Permasalahan yang ada dalam sistem donasi saat ini adalah kurangnya transparansi pada proses donasi mulai dari penerimaan donasi dari para donatur ke lembaga, proses pendistribusian, hingga donasi fiktif. Permasalahan ini muncul karena beberapa oknum yang mencoba mencari keuntungan untuk diri sendiri sehingga para donatur yang ingin membantu sesama mulai kehilangan kepercayaan kepada lembaga donasi yang ada.

#### 3.1.2 Analisis dan Pemecahan Masalah

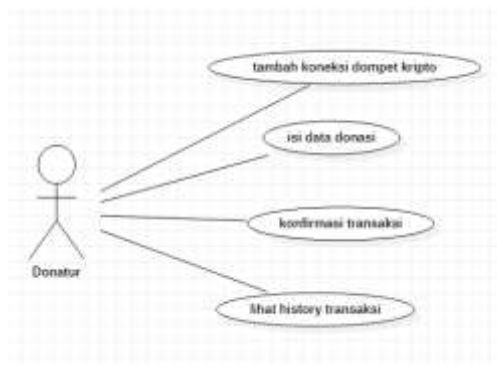
Sebagai salah satu cara untuk mengatasi permasalahan tranparansi donasi, blockchain dapat digunakan untuk mengatasi permasalahan tersebut. Blockchain hadir dengan sistem buku besar *read and write* dimana pembukuan pada proses donasi dapat tercatat secara akurat dan tidak dapat dihapus ataupun diubah. Blockchain tidak seperti sistem

aplikasi lain yang basis datanya dapat dikendalikan sepenuhnya oleh pemilik server. Setiap orang yang menggunakan blockchain dapat menjadi pemegang server. Dengan sistem blockchain yang dapat diakses oleh siapa saja, maka dapat membangun rasa kepercayaan para donatur kembali.

### 3.2 Perancangan Sistem

Perancangan perangkat lunak meliputi pembuatan *smart contract* dan pembuatan antarmuka web. Perangkat lunak memiliki fitur transaksi yang merupakan fitur untuk melakukan donasi yang selanjutnya melakukan operasi *create* pada blockchain sebagai pencatatan transaksi donasi dan operasi *read* yang digunakan pada antarmuka web untuk menampilkan riwayat transaksi.

#### 3.2.1 Use Case Diagram



Gambar 1. Use Case Diagram

Pada Gambar 1 menunjukkan rancangan use case diagram untuk aplikasi donasi terdesentralisasi.

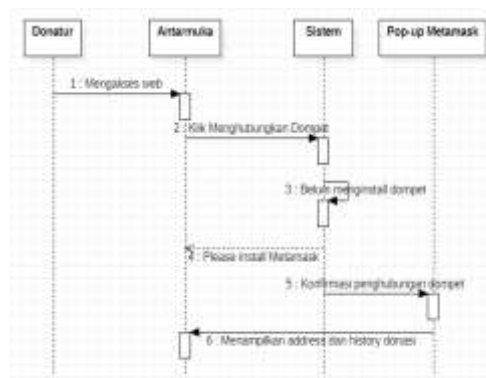
Tabel 1. Deskripsi Use Case

No	Aktivitas	Deskripsi
1	Hubungkan Dompet	Donatur menghubungkan dompet crypto ke website yang akan digunakan sebagai media pembayaran
2	Donasi	Donatur melakukan donasi dengan cara mengisi form donasi

3	Konfirmasi	Donatur melakukan konfirmasi untuk donasi sebesar nilai yang didonasikan
4	Lihat Riwayat Transaksi	Donatur melihat riwayat transaksi

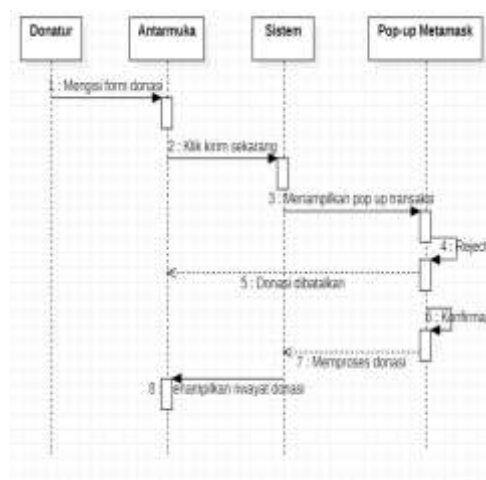
Tabel 1 menunjukkan deskripsi dari use case.

#### 3.2.2 Sequence Diagram



Gambar 2. Sequence Diagram Menghubungkan Dompet

Pada Gambar 2 menunjukkan sequence diagram untuk menghubungkan dompet kripto pada aplikasi donasi terdesentralisasi.

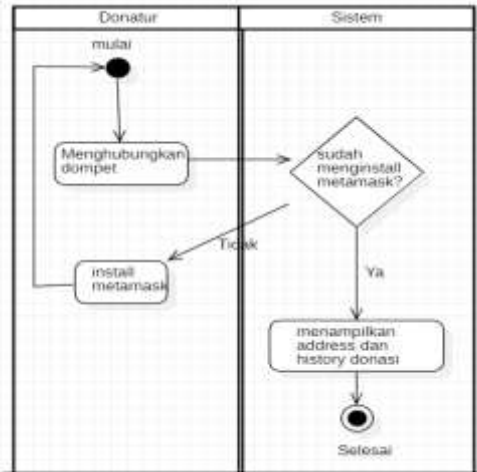


Gambar 3. Sequence Diagram Donasi

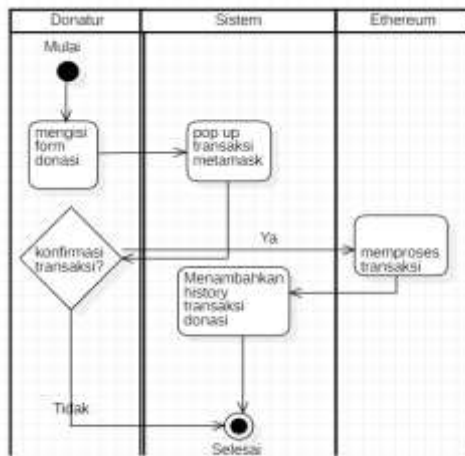
Gambar 3 adalah sequence diagram untuk melakukan donasi.

### 3.2.3 Activity Diagram

Gambar 4 dan Gambar 5 adalah activity diagram untuk proses menghubungkan dompet dan melakukan donasi.



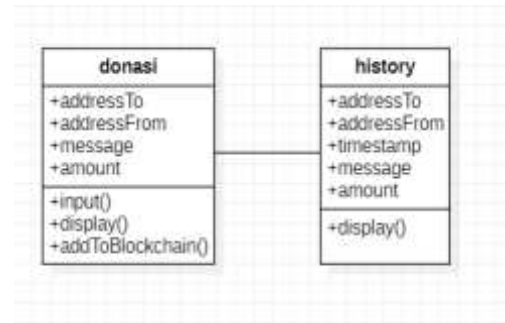
Gambar 4. Activity Diagram Menghubungkan Dompet



Gambar 5. Activity Diagram Donasi

### 3.2.4 Class Diagram

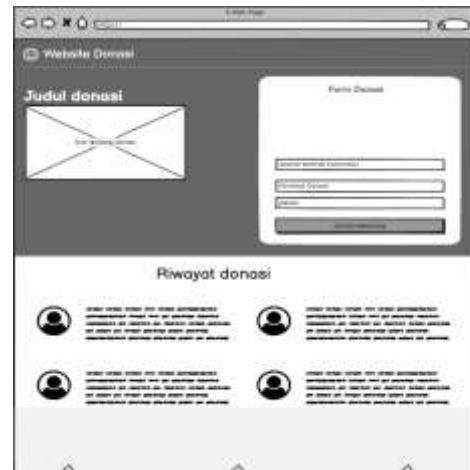
Gambar 6 merupakan rancangan class diagram untuk aplikasi donasi terdesentralisasi.



Gambar 6. Class Diagram

### 3.2.5 Antarmuka

Rancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga mempermudah pembuatan aplikasi. Aplikasi web ini juga berfungsi sebagai penghubung antara pengguna dalam hal ini donatur dengan *smart contract*.



Gambar 7. Antarmuka

Pada Gambar 7 menunjukkan rancangan antarmuka untuk aplikasi donasi terdesentralisasi.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Implementasi Sistem

Implementasi sistem berguna untuk mengetahui dan memastikan fungsional pada aplikasi telah berjalan sesuai dengan yang diharapkan dan untuk mengetahui kesalahan-kesalahan yang terdapat pada aplikasi.

## 4.2 Migrasi dan Pengujian Smart Contract

### 4.2.1 Migrasi Smart Contract

Proses migrasi adalah proses *deployment* dari *smart contract* yang telah dibuat dan disebarakan menuju jaringan blockchain. Migrasi *smart contract* dilakukan dengan Remix Ethereum IDE. *Smart contract* dimigrasikan ke dalam jaringan Testnet Rinkeby.

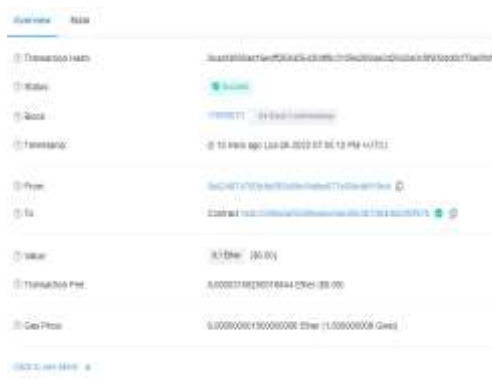


Gambar 8. Proses Migrasi Smart Contract

Pada Gambar 8 menunjukkan bahwa proses migrasi *smart contract* ke jaringan Testnet Rinkeby telah berhasil dilakukan dan didapatkan alamat *smart contract* yang akan digunakan, yaitu alamat hash seperti berikut ini 0x6c3399a2a56396ea6e2ae4563b7d84dd2df2f97b

### 4.2.2 Pengujian Fungsi Smart Contract

Pengujian ini dilakukan untuk mencoba semua fungsi yang telah dibuat pada *smart contract*. Untuk lebih memperjelas tiap fungsi yang telah dibuat, berikut gambar beserta penggalan script dan penjelasannya.



Gambar 9. Fungsi External Payable

```

constructor() {
    owner = payable(msg.sender);
}

receive() external payable {}
    
```

Fungsi dari external payable yang dibuat adalah agar *smart contract* tidak hanya melakukan perintah tetapi juga dapat menampung saldo donasi.



Gambar 10. Fungsi addToBlockchain

```

function addToBlockchain(address payable receiver, uint amount, string memory message, string memory keyword) public {
    transactionCount += 1;
    transactions.push(TransferStruct(msg.sender, receiver, amount, message, block.timestamp, keyword));
}
    
```

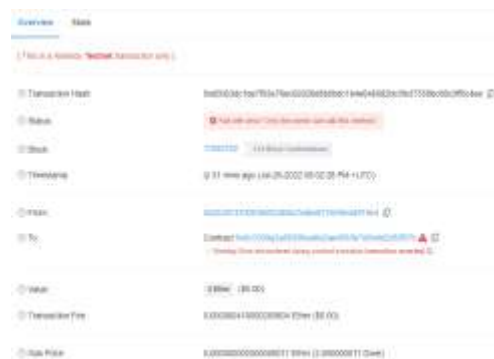
```
emit Transfer(msg.sender, receiver,
amount, message, block.timestamp,
keyword);
}
```

Fungsi ini dibuat sebagai pembukuan riwayat transaksi donasi yang disimpan pada blockchain.



Gambar 11. Fungsi Withdraw Berhasil

Gambar 12 menunjukkan pemanggilan fungsi berhasil dilakukan.



Gambar 12. Fungsi Withdraw Gagal

```
function withdraw(uint _amount) external
{
require(msg.sender == owner, "Only
the owner can call this method.");
payable(msg.sender).transfer(_amount
);
}
```

Fungsi withdraw berfungsi untuk menarik saldo hasil donasi. Fungsi ini hanya bisa dieksekusi oleh pemilik *smart contract* agar fungsi ini tidak disalahgunakan oleh orang yang tidak bertanggung jawab. Pada Gambar 12 menampilkan hasil pemanggilan fungsi yang gagal dikarenakan hanya pembuat *smart contract* saja yang dapat menggunakan fungsi tersebut.



Gambar 13. Fungsi GetAllTransactions

```
function getAllTransactions() public view
returns (TransferStruct[] memory) {
return transactions;
}
```

Fungsi ini digunakan untuk menampilkan semua riwayat yang disimpan oleh fungsi *addToBlockchain* yang akan dipanggil pada antarmuka web.

### 4.3 Tampilan Aplikasi



Gambar 14. Tampilan Antarmuka Aplikasi Web

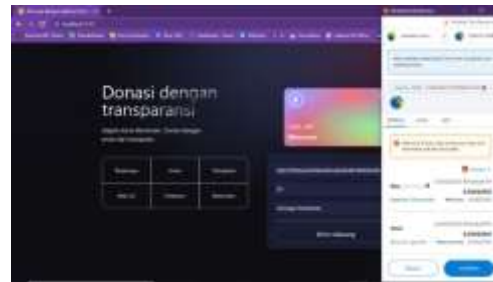
Gambar 14 adalah tampilan antarmuka aplikasi web yang telah dibangun, dimana pada saat mengakses web pertama kali akan muncul notifikasi untuk mengunduh Metamask jika pengguna belum memilikinya.





Gambar 15. Tampilan Setelah Menghubungkan Dompet Kripto (Metamask)

Setelah menghubungkan dompet kripto Metamask, tampilan pada kotak dompet akan memunculkan alamat Ethereum dan sudah siap untuk melakukan donasi seperti pada Gambar 15.



Gambar 18. Proses Pembukuan Blockchain

Setelah melakukan konfirmasi transaksi donasi, akan muncul *pop up* kedua, seperti yang terlihat pada Gambar 18, sebagai bagian dari fungsi pembukuan *addtoblockchain*. Fungsi ini berguna untuk mencatat isian form ke dalam blockchain.



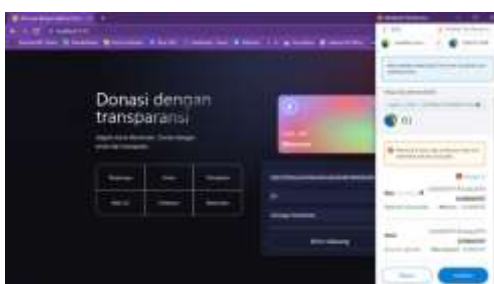
Gambar 16. Tampilan Informasi Donasi

Pada Gambar 16 menunjukkan informasi donasi yang telah dilakukan.



Gambar 19. Tampilan Riwayat Transaksi

Seluruh riwayat transaksi yang telah berhasil dilakukan akan tampil pada aplikasi seperti yang terlihat pada Gambar 19.



Gambar 17. Proses Donasi

Setelah mengisi form donasi akan muncul *pop up* dari Metamask yang berisi tentang informasi transaksi seperti alamat tujuan, nominal dan *gas fee* atau biaya seperti yang terlihat pada Gambar 17.

## 5. KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan, dapat disimpulkan bahwa aplikasi donasi berbasis blockchain telah berhasil dibangun sesuai dengan tujuan dari penelitian ini. Donatur dapat melihat langsung transaksi yang terjadi pada *smart contract* sehingga para donatur menjadi lebih percaya. Aplikasi ini juga dapat menampilkan riwayat donasi sehingga menjadi lebih transparan dan publik dapat memantau secara langsung.

## DAFTAR PUSTAKA

- Prashanth Joshi, A., Han, M., & Wang, Y. (2018). A Survey on Security and Privacy Issues of Blockchain Technology. *Mathematical Foundations of Computing*, 1(2), 126. <https://doi.org/10.3934/mfc.2018007>
- Luu, L., Chu, D. H., Olickel, H., Saxena, P., & Hobor, A. (2016). Making Smart Contracts Smarter. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (pp. 254-269).
- Voshmgir, S. (2019). *Token Economy: How Blockchains and Smart Contracts Revolutionize the Economy*. Shermin Voshmgir-BlockchainHub.
- Yaga, D., Mell, P., Roby, N., & Scarfone, K. (2019). *Blockchain Technology Overview*. arXiv Preprint arXiv:1906.11078.
- Iswari, L. (2021). Penerapan React JS Pada Pengembangan FrontEnd Aplikasi Startup Ubaform. *AUTOMATA*, 2(2).