CO3302 Computer Engineering Project

Blockchain Transaction System

Chanvitha Praveen

October 9, 2022

Abstract—This project is totally based on blockchain based WEB3.0 technology. Due to the traditional problems in fiat currencies such as high inflation, low transparency, low security, centralized authority to the government, etc. nowadays digital currencies are much popular than fiat currencies. Digital currencies are totally decentralized, much secure, pure transparency, has low fees. As an example, Bitcoin, Ethereum etc. When using digital currencies, it needs to have a digital platform to transfer and receive money from other parties. So, "Cryptanz" is the platform that is going to build to transfer digital currencies from one party to another party via Ethereum blockchain. In this project we use Ethereum chain but Rapsan-Test network to implement it. Because otherwise we have to pay unnecessary gas fees due to testing. This platform can view all the transactions has done clearly in "Etherscan", but as a public address, not by any kind of user credentials. Also, the amount of digital currency that we have is stored in "Metamask" digital wallet integrated to this platform. Hence this project is a digital currency transaction platform, totally based on smart contract of blockchain technology.

Contents

1 Introduction

Tim Berners-Lee unveiled the world's first web browser in 1993. This was the first version of the internet. It provided a new avenue for individuals all around the world to get information, but navigation and design were difficult to master. Then came sites like Google, Yahoo, and Facebook. They compiled information and made searching a breeze. This was the Web 2.0 era. However, many people today believe that these internet behemoths have too much power. Web3 was born as a result. The next edition of the web, dubbed Web3, focuses on decentralization, which will ultimately return control to the end-user. Web3 provides services in the form of decentralized apps (dapps) that run on the blockchain network. Ethereum is perhaps the most well-known of these networks. [1] As a result, the digital currency much popular community wanted a platform to do transactions in digital currencies. So, the proposed project is for that occurred problem. This project can do transactions on the Ethereum blockchain and give a clear transparency of all the transactions have done through. All the transactions guarantee the security because the platform integrated with the "Metamask" private key handled wallet. Since the funds can operate by the original user only. What is blockchain? A Blockchain represents a distributed ledger of transactions running in a peer-to-peer network, where those transactions, once confirmed, cannot be deleted or modified.

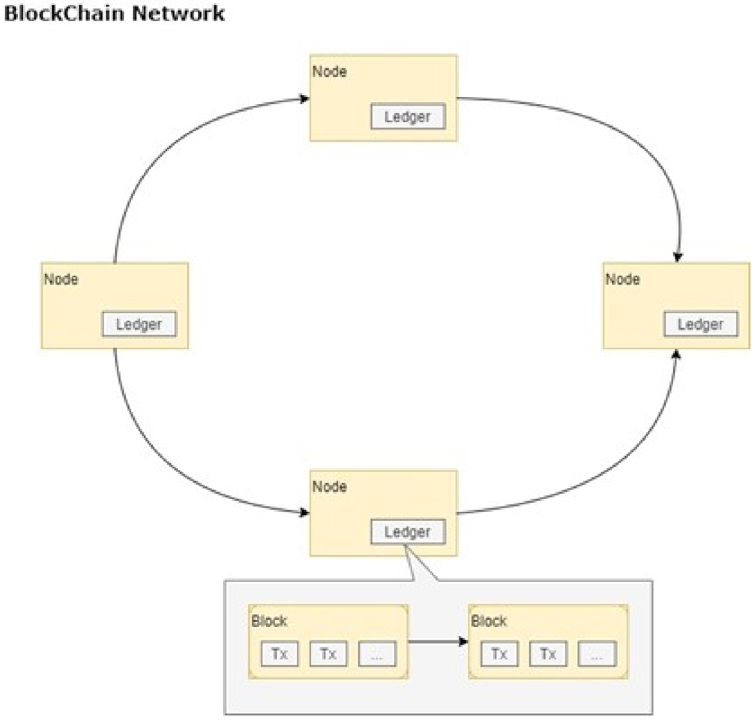
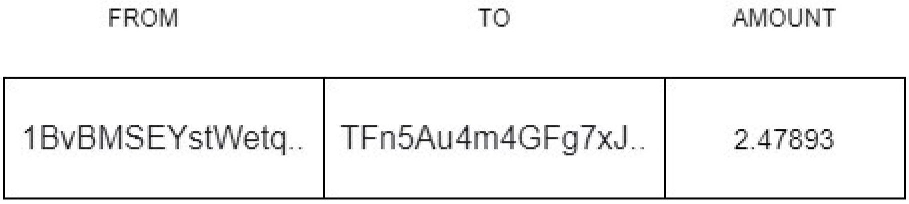


Figure 1: Blockchain Network (Source: Adapted from [2])

The image above gives a visual representation of the Blockchain so we can grasp the concept. We can see it as transactions grouped in blocks, which are chained together. The chain is a logical pointer from one block to the previous one. As it would happen with any peer-to-peer network you might use in the past for sharing files, the same concepts apply to Blockchain. It does not have any central authority or server, and all the participants in the network replicate the ledger of transactions. The participants who join the network for submitting transactions or reading data from the ledger are known as Nodes. Last but not least important, most Blockchains are public, making the information on them available on-demand for everyone and representing a single source of truth. (Transparency)



FROM	TO	AMOUNT
1BvBMSEYstWetq..	TFn5Au4m4GFg7xJ..	2.47893

Figure 2: Sample Transaction (Source: Adapted from [2])

2 Problem Definition

2.1 Ending of Web 2.0

As a result of ending the web 2.0 the blockchain technology with web 3.0 is much popular. Hence the background for a new platform has more opportunity in the market.

2.2 People need transparency of money transactions

Every country has this issue, that there is no transparency of the money flow. But in digital currency transaction system via blockchain system has high transparency.

2.3 Inflation of fiat currencies

Due to world pandemics, world wars, weak leaders the money inflation is coming into the picture. And huge amount of people are tried to adapt with digital currencies not to reduce their monitory value of money and easy transfer via internet.

2.4 Can Include more functionalities

Not only digital currency transaction system, but also additionally it can include a private NFT clone as a further improvement in future. Then users can mint their own NFT's in this platform.

3 Objectives

- Create the front end with a stunning design to user friendly
- Create "Metamask" wallet by using testing Ethereum in Rapsan-Test network
- Develop the smart contract by connecting to the blockchain
- Integrate the smart contract to the front-end
- Test the system whether transactions are done properly
- Show the latest transactions in the front-end area with a simple gif
- Verify the transactions by scanning through "Etherscan"
- Additionally hope to add personal NFT clone with integrated "Opensea" platform

4 Milestones

- Requirement analysis
- Architectural design
- Develop front-end
- Develop smart contract
- Connect to the blockchain
- Testing
- Deployment

5 Methodology

For this Application develop process, the Agile technique is used because if there are some capabilities with the newer solidity versions, then the application can be optimized.

5.1 Requirement Analysis

This stage is the most valuable stage that we can consider before the project start. Because we have to conduct a feasibility case study that we will not get any trouble or problems in future when we started to develop the proposed project. Both software tools and hardware tools configurations should be analyzed at this stage.

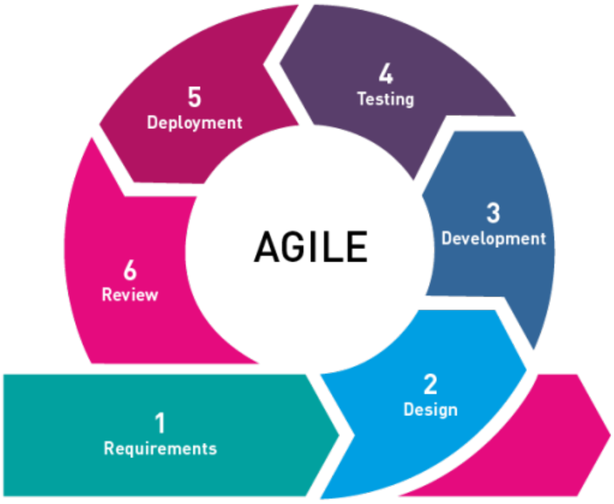


Figure 3: Agile Method Development

5.2 Architectural Design

During this stage, the overall architecture of the design process as well as the components of the proposed project are designed. The essential software and hardware modules and components must be designed, and their interactions must be matched.

5.3 Plan and develop a stunning front-end design

Before starting to develop the front end we should make a sketch of the front end. Otherwise, it will not neat and efficient after integrating to the smart contract.

5.4 Create smart contract

The backend part is done in this stage. Solidity language is used for this and develop the functionalities of the transaction section. This should be more powerful and efficient when doing transactions. For that we use some new environments such as “Hardhat”, “Alchemy” etc.

5.5 Front-end connect with the smart contract

At this stage we have to integrate both sections that we developed separately into one project. In this case there might have more problems, but we have to solve them. [?]

5.6 Security Implementations of the System

5.6.1 PSQL Database

- This database used in Supabase platform to implement in the system. This provides the additional security for login rather than Google Sign-In method. Because it has Pro Level Security, and it gives the system not to access others rather than the own user account.

5.6.2 Google Sign-In

- In here used the Google-Sign-in system to sign in to the system, but as a future improvement hope to add google authentication two step verification to the system to increase the security of the system.

5.7 Connect Wallet to the web dapp

In here the “Metamask” wallet is connected to the decentralized web application and get some test Ethereum to the wallet via “Rapsan-Test” network

5.8 View latest transactions

In bottom page of the web application there will have a section that all the transactions through our wallet. All these information are public and anyone can view and verify via “Etherscan” if they click on a transaction.

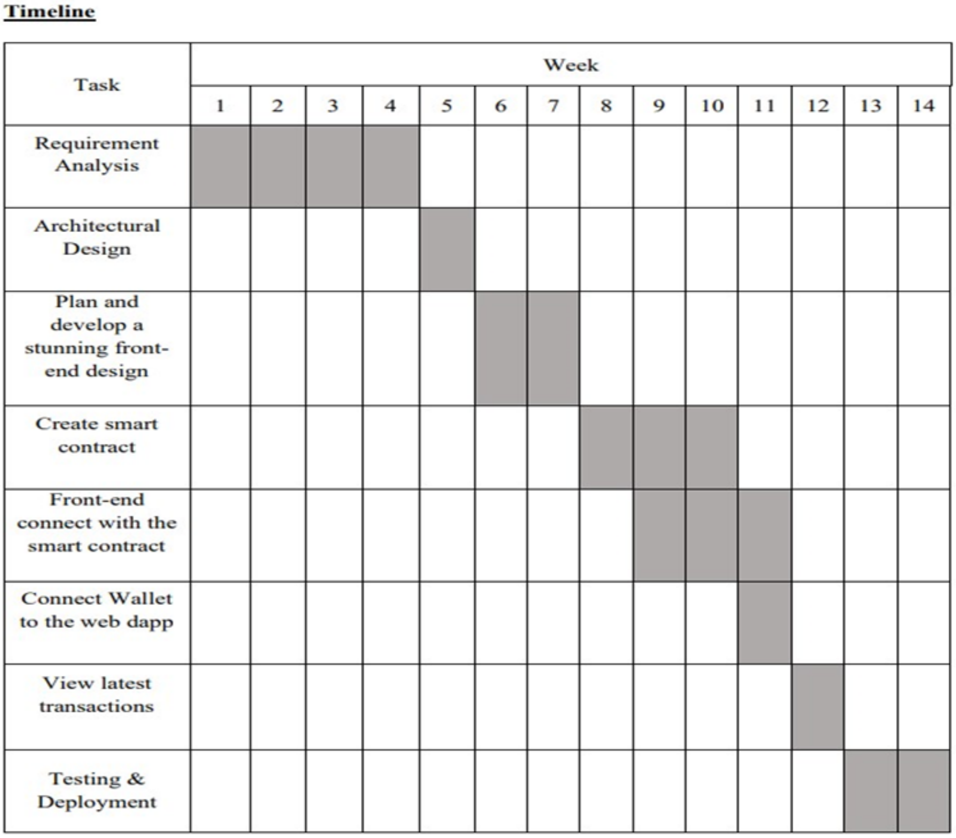
5.9 Testing

In V model time to time the unit testing is done. And finally whole system testing is done. In here after connecting front-end to the smart contract and after testing other all functionalities, the whole testing can be done.

5.10 Deployment

After completing the project in our localhost, we can deploy it to the public in the Internet. We need a domain and hosting to do that. But at this stage we might not go over than the localhost.

6 Timeline



Timeline

Task	Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Requirement Analysis	▨	▨	▨	▨										
Architectural Design					▨									
Plan and develop a stunning front-end design						▨	▨							
Create smart contract								▨	▨	▨				
Front-end connect with the smart contract									▨	▨	▨			
Connect Wallet to the web dapp											▨			
View latest transactions												▨		
Testing & Deployment													▨	▨

Figure 4: Timeline of the Project

7 Tools and Resources

7.1 Tools for front-end

- react.js
- “Vite” is used to initialize the react project easily [5]
- “Tailwindcss” is used to style the web easily in react.js [6]

7.2 Tools for smart contract

- Solidity programming language
- “Hardhatis” used for Ethereum development environment, and it allows to run solidity locally

- “Tailwindcss” is used to style the web easily in react.js [6]
- “Ropsten testnet” faucet website is used to get free test Ethereum because unless we have to pay Ethereum gas fees
- “Metamask” wallet extension
- “Alchemy” website is used to deploy easily the test project
- “Giffy” developers web site s used to show interactive gifs in latest transactions

8 Description

8.1 Login System

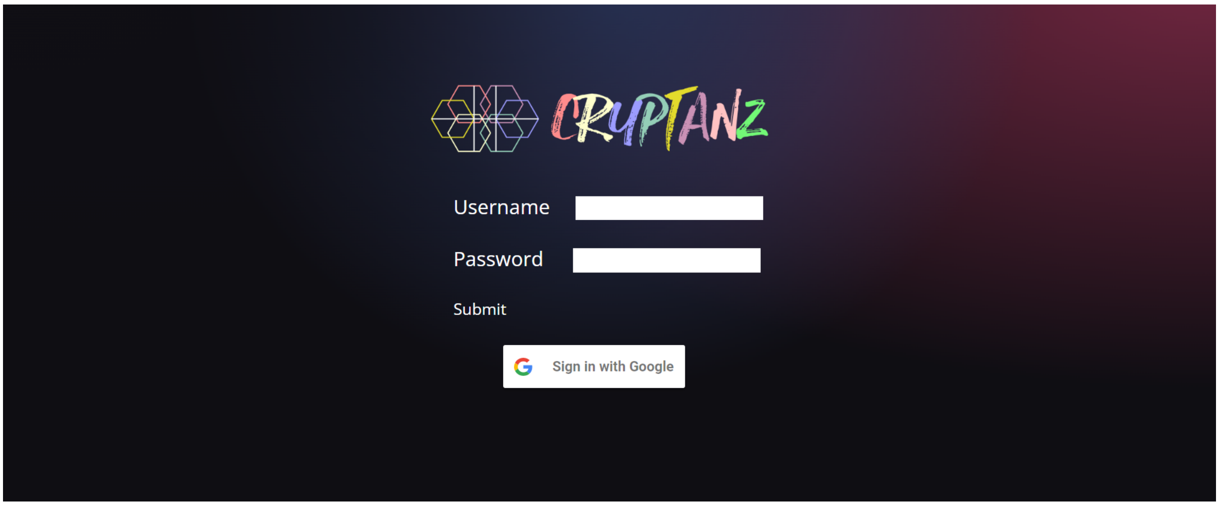
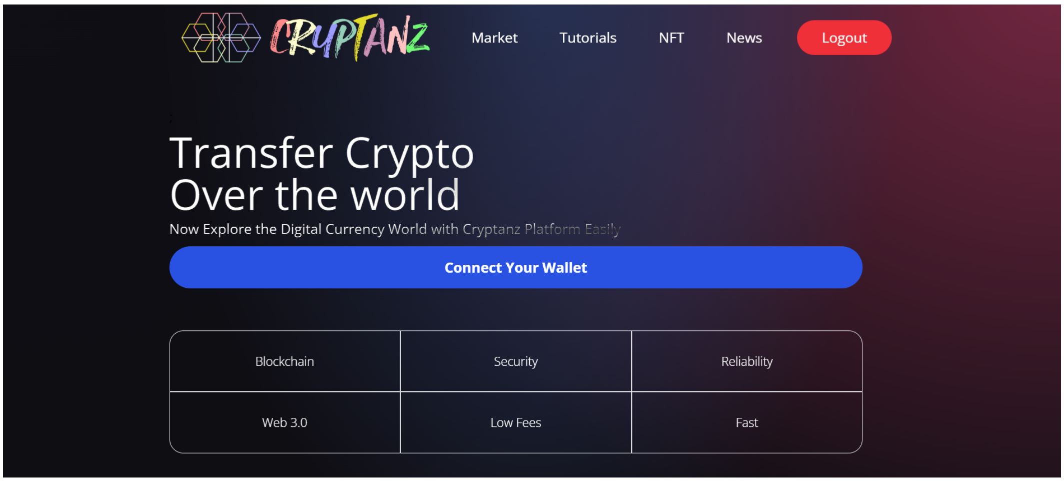


Figure 5: Login of the System

In the logging system, the user can log via using his/her username and password or directly by using the google sign-in method. The logging details is using via supabase database.

8.2 Home Page



Transfer Crypto
Over the world

Now Explore the Digital Currency World with Cryptanz Platform Easily

Figure 6: Home Page of the System

This is the home page of the dapp system and in here you can see some of hyperlinks to different sections in the navigation bar. And also you can connect your wallet using the connect your wallet button. Also you can logout if you want to .

8.3 Transferring System

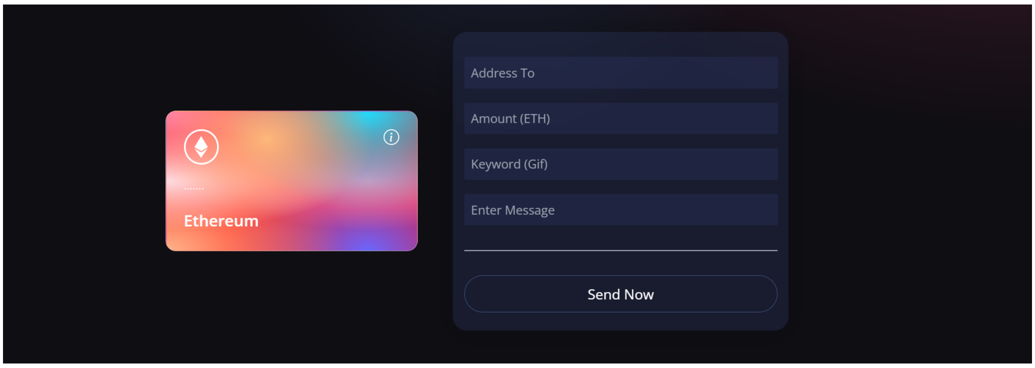
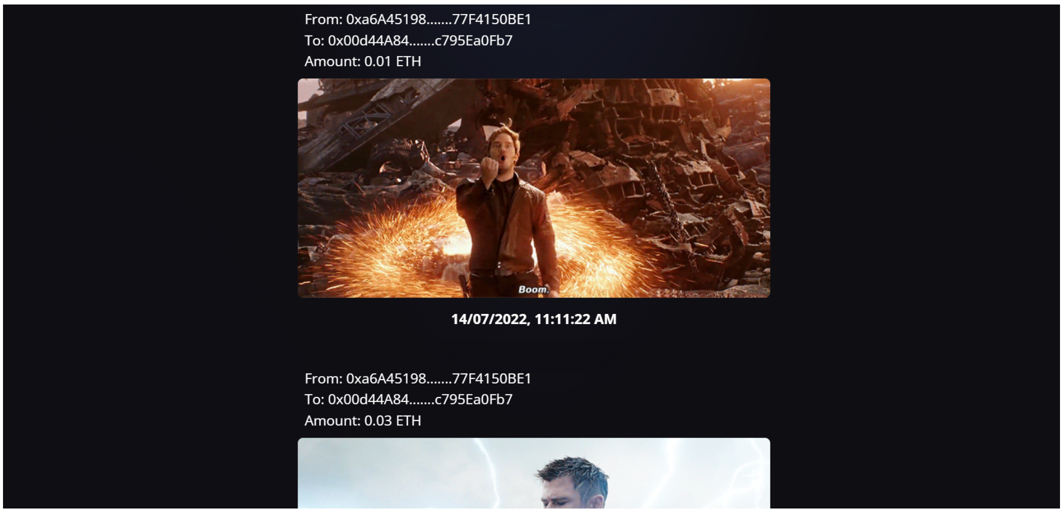


Figure 7: Digital Assets Transferring GUI of the System

In here the main section of the system and, you can enter the required details then the block chain transaction is done by this section. The sending time is preview by a circulating indicator when the send now button clicked.

8.4 Latest Transactions



From: 0xa6A45198.......77F4150BE1
To: 0x00d44A84.......c795Ea0Fb7
Amount: 0.01 ETH

14/07/2022, 11:11:22 AM

From: 0xa6A45198.......77F4150BE1
To: 0x00d44A84.......c795Ea0Fb7
Amount: 0.03 ETH

Figure 8: Latest Transactions Section of the System

In here the latest transactions are displayed. Can be refer to the transparency of transactions via Etherscan by clicking on the details.

9 Cost Analysis / Budget

In this project we do not have to spend much cost because we do the project on software-based system. But if we deploy this project to public internet, we need some of spends to full fill the project.

Domain name - $10.00

For Hosting - $35.00 per year

Get some Ethereum - $50.00

Hence the total cost might be $95.00

10 Risks

Risks can be divided into two categories. The below information is about some risks and what are the alternative paths to achieve the goals even there is a risk.

10.1 When the project is developing

- The solidity programming language is rapidly updating with newer versions

Can be stick with a static version even the solidity rapidly changing.

- Getting various errors when developing the project

  Refer to the internet and ask some expert person in this field. (Some contacts are available to get help) [4]

- Challenge to make the decentralized web application more efficiently

  Follow short-circuiting, remove unnecessary libraries, use proper data types, refer gas-costly patterns, not use expensive loops, etc. [3]

10.2 After the project developed

- Getting some problems while testing

- Challenge to build a creative market

  User can do several activities in one platform. Transfer money to another person's wallet, send some message to the user with security feature (AES), verify the public addresses and see the latest transactions in the sale page

11 Conclusion

Develop a system or a product to solve a human problem or a problem that facing humans nowadays is called as an engineering solution for a existing problem. In this case the problem is to hard to find all services which want to transfer and verify digital currency transactions in a same platform. Hence the "Cryptanz" platform is the solution for that. In normal way people can use digital wallet separately, send messages in a separate platform, view all the transactions in a different platform, etc. But in this project people can do all the services that mentioned above in one platform. That is the easiness and innovation built for the market.

Additionally we hope to add another service to this platform which is personal NFT clone integrated with "Opensea" platform. Then user can promote their NFT's in the same platform by sharing the @username of NFT clone. And also hope to add PGP mail server sign-in option rather than Google sign-in method. Also hope to develop the encryption message system by using AES and it will be a huge improvement in this system. The total project will be built using the Agile Method, which is one of the most widely used software engineering models for project development.

12 Appendix

References

[1] Integrate your react app with smart contracts - dev community , NaN.

[2] ”101 Smart Contracts and Decentralized Apps in Ethereum”, Auth0 - Blog, 2022. [Online]. Available: https://auth0.com/blog/101-smart-contracts-and-decentralized-apps-in-ethereum/. [Accessed: 22- Mar- 2022]

[3] ”How to Write Smart Contracts That Optimize Gas Spent on Ethereum”, Medium, 2022. [Online]. Available: https://betterprogramming.pub/how-to-write-smart-contracts-that-optimize-gas-spent-on-ethereum-30b5e9c5db85. [Accessed: 22- Mar- 2022]

[4] ”Solidity”, DEV Community, 2022. [Online]. Available: https://dev.to/t/solidity. [Accessed: 22- Mar- 2022]

[5] ”Creating a React.js App using Vite”, Engineering Education (EngEd) Program — Section, 2022. [Online]. Available: https://www.section.io/engineering-education/creating-a-react-app-using-vite/. [Accessed: 22- Mar- 2022]

[6] ”Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.”, Tailwindcss.com, 2022. [Online]. Available: https://tailwindcss.com. [Accessed: 22- Mar- 2022]

[7] ”Integrate your React App with Smart Contracts”, DEV Community, 2022. [Online]. Available: https://dev.to/nazeeh21/integrate-your-react-app-with-smart-contracts-4o3m. [Accessed: 22- Mar- 2022]