Index

ABSTRACT

A cryptocurrency, crypto-currency, crypto, or coin is a digital currency designed to work as a medium of exchange through a computer network that is not reliant on any central authority, such as a government or bank, to uphold or maintain it.

We define a any cryptocurrency as a chain of digital signatures. Each owner transfers it to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin. A payee can verify the signatures to verify the chain of ownership.

Our web application helps transfer Ethereum to another wallet on the ERC20 network in using minimal gas fees. We show all the transactions on the blockchain using a GIFHY Api that takes a tag and a message.

Tools Used

React.js

Solidity

HardHat

MetaMask

ViteApp

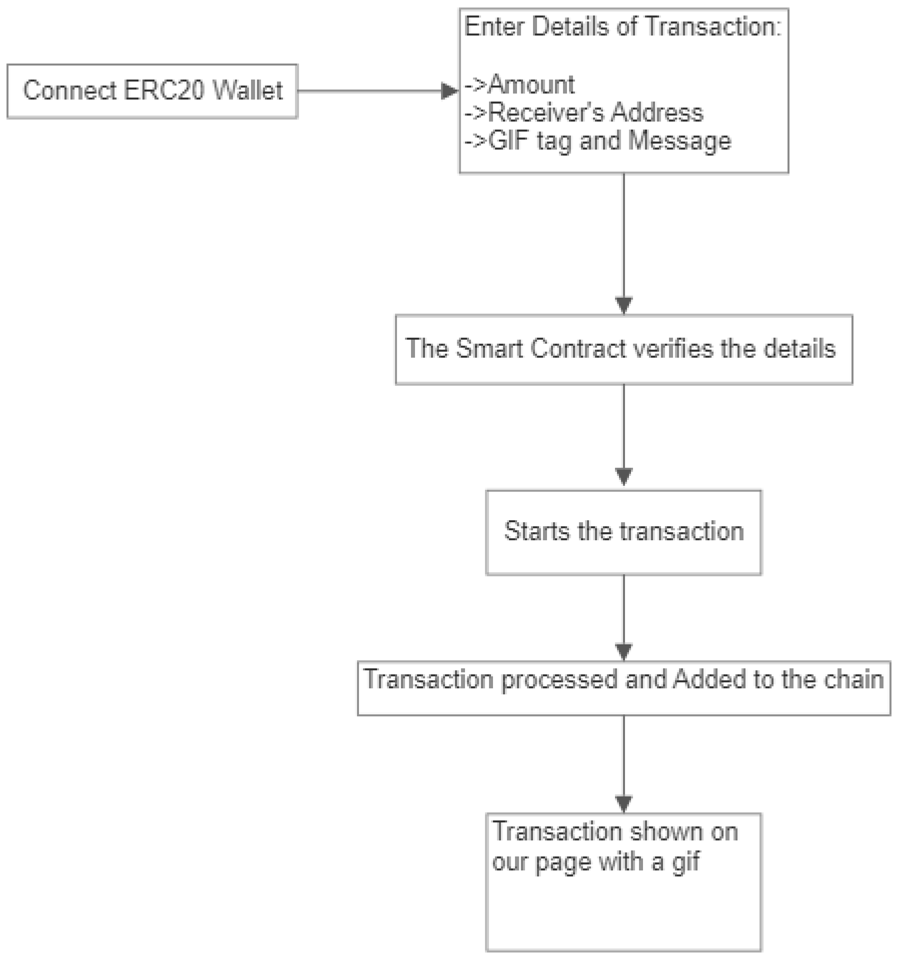
HTML

Tailwind.CSS

Node.js

GIPHY API

Block Diagram


Connect ERC20 Wallet
Enter Details of Transaction:
->Amount
->Receiver's Address
->GIF tag and Message
The Smart Contract verifies the details
Starts the transaction
Transaction processed and Added to the chain
Transaction shown on our page with a gif

Project Modules

Frontend:

The frontend of our website provides the GUI for all the transaction requests, connections and for displaying all the transactions that exist for that wallet. The frontend of the website is built using React.js, HTML, Tailwind.CSS. We use Vite App for rendering the frontend.

Transaction Form:

The transaction form takes in the receiver’s address, amount, gif keyword and message.

Transactions:

Displays all the transactions stored on Etherscan along with the giphy keyword.

Backend / Blockchain:

Blockchain is a system of recording information in a way that makes it difficult or impossible to change, hack, or cheat the system.

A blockchain is essentially a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. Each block in the chain contains a number of transactions, and every time a new transaction occurs on the blockchain, a record of that transaction is added to every participant’s ledger. The decentralised database managed by multiple participants is known as Distributed Ledger Technology

Direct-X Crypto Currency transfer system is built on Ethereum block chain and tested on Ropsten test network. The connect wallet button connects the wallet to the website and upon entering the receiver’s address, the amount of ETH to be sent, a message and a keyword, the website will generate a GIF that best fits the keyword, and will store the transaction on the block chain and show the transaction with the gif on the website.

Code Features

React.js:

Making the frontend and building user interfaces specifically for the transactions. It’s also useful as we can make reusable components for the page

Json:

For storing all the dependencies for the smart contracts and gif keywords with all the wallet ids, messages and amount

Tailwind.CSS:

For making all the gradients and cards and design preferences in the website.

Flex:

For asynchronously updating the transactions eliminating the need for refreshing the site again and again.

Sample Codes And Outputs

Welcome.jsx

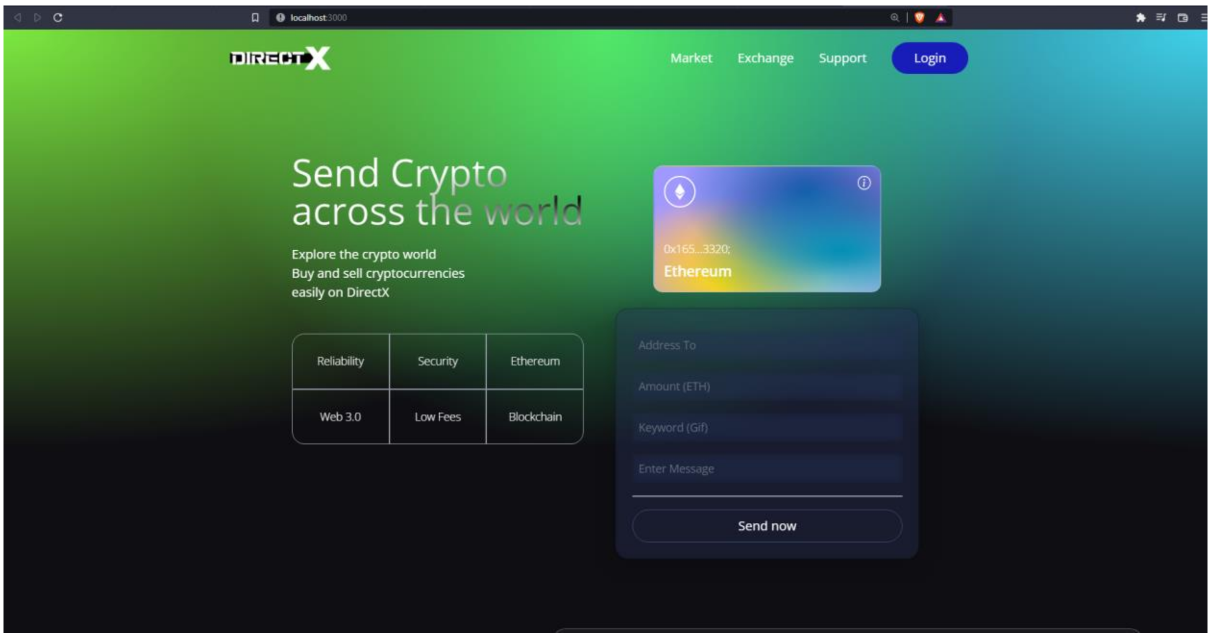
```jsx
import React, { useContext } from "react";
import { AiFillPlayCircle } from "react-icons/ai";
import { SiEthereum } from "react-icons/si";
import { BsInfoCircle } from "react-icons/bs";

import { TransactionContext } from "../context/TransactionContext";
import { Loader } from "./";
import { shortenAddress } from "../utils/shortenAddress";

const companyCommonStyles =
  "min-h-[70px] sm:px-0 px-2 sm:min-w-[120px] flex justify-center items-center
border-[0.5px] border-gray-400 text-sm font-light text-white";

const Input = ({ placeholder, name, type, value, handleChange }) => (
  <input
    placeholder={placeholder}
    type={type}
    step="0.0001"
    value={value}
    onChange={(e) => handleChange(e, name)}
    className="my-2 w-full rounded-sm p-2 outline-none bg-transparent text-
white border-none text-sm white-glassmorphism"
  />
);

const Welcome = () => {
  const {
    connectWallet,
    currentAccount,
    formData,
    handleChange,
    sendTransaction,
    transactions,

    isLoading,
  } = useContext(TransactionContext);

  const handleSubmit = (e) => {
    const { addressTo, amount, keyword, message } = formData;

    e.preventDefault();
    if (!addressTo || !amount || !keyword || !message) return;

    sendTransaction();
  };

  return (
    <div className="flex w-full justify-center items-center">
      <div className="flex mf:flex-row flex-col items-start justify-between
md:p-20 py-12 px-4">
        <div className="flex flex-1 justify-start items-start flex-col mf:mr-
10">
          <h1 className="text-3xl sm:text-5xl text-white text-gradient py-1">
            Send Crypto <br /> across the world
          </h1>
          <p className="text-left mt-4 text-white font-heavy md:w-8/12 w-11/12
text-base">
            Explore the crypto world
            <br /> Buy and sell cryptocurrencies easily on DirectX
          </p>
          {!currentAccount && (
            <button
              type="button"
              onClick={connectWallet}
              className="flex flex-row justify-center items-center my-5 bg-
[#181dba] p-3 rounded-full curson-pointer hover:bg-[#0E8FAC]"
            >
              <p className="text-white text-base font-semibold">
                Connect Wallet
              </p>
            </button>
          )}

          <div className="grid sm:grid-cols-3 grid-cols-2 w-full mt-10">
            <div className={`rounded-tl-2xl ${companyCommonStyles}`}>
              Reliability
            </div>
            <div className={companyCommonStyles}>Security</div>
            <div className={`sm:rounded-tr-2xl ${companyCommonStyles}`}>
              Ethereum
            </div>
            <div className={`sm:rounded-bl-2xl ${companyCommonStyles}`}>
              Web 3.0
            </div>
            <div className={companyCommonStyles}>Low Fees</div>
            <div className={`rounded-br-2xl ${companyCommonStyles}`}>
              Blockchain
            </div>
          </div>
        </div>

        <div className="flex flex-col flex-1 items-center justify-start w-full
mf:mt-0 mt-10">
          <div className="p-3 justify-end items-start flex-col rounded-xl h-40
sm:w-72 w-full my-5 eth-card white-glassmorphism">
            <div className="flex justify-between flex-col w-full h-full">
              <div className="flex justify-between items-start">
                <div className="w-10 h-10 rounded-full border-2 border-white
flex justify-center items-center">
                  <SiEthereum fontSize={21} color="#fff" />
                </div>
                <BsInfoCircle fontSize={17} color="#fff" />
              </div>
              <div>
                <p className="text-white font-light text-sm">
                  {shortenAddress(currentAccount)};
                </p>
                <p className="text-white font-semibold text-lg mt-1">
                  Ethereum
                </p>
              </div>
            </div>
          </div>

          <div className="p-5 sm:w-96 w-full flex flex-col justify-start
items-center blue-glassmorphism">
            <Input
              placeholder="Address To"
              name="addressTo"
              type="text"
              handleChange={handleChange}
            />
            <Input
              placeholder="Amount (ETH)"
              name="amount"
              type="number"
              handleChange={handleChange}
            />
            <Input
              placeholder="Keyword (Gif)"
              name="keyword"
              type="text"
              handleChange={handleChange}
            />
            <Input
              placeholder="Enter Message"
              name="message"
              type="text"
              handleChange={handleChange}
            />

            <div className="h-[1px] w-full bg-gray-400 my-2" />

            {isLoading ? (
              <Loader />
            ) : (
              <button
                type="button"
                onClick={handleSubmit}
                className="text-white w-full mt-2 border-[1px] p-2 border-
[#3d4f7c] hover:bg-[#3d4f7c] rounded-full cursor-pointer"
              >
                Send now
              </button>
            )}
          </div>
        </div>
      </div>
    </div>
  );
};

export default Welcome;
```

Contracts.sol

```
//SPDX-License-Identifier: UNLICENSED


pragma solidity ^0.8.0;


contract Transactions{
    uint256 transactionCount;


    event Transfer(address from,address receiver,uint amount,string
message,uint256 timestamp,string keyword);
```

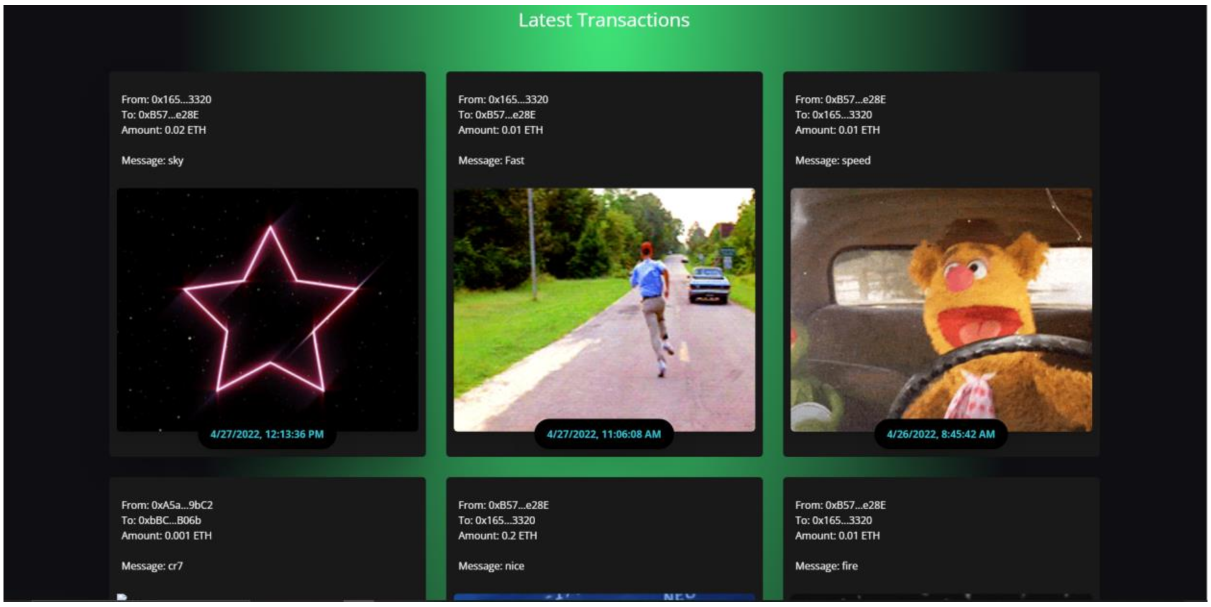
```
  struct TransferStruct{
      address sender;
      address receiver;
      uint amount;
      string message;
      uint256 timestamp;
      string keyword;
  }

  TransferStruct[] transactions;
  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);
  }

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

  function getTransactionCount() public view returns (uint256){
      return transactionCount;
  }


}
```

Output:



Send Crypto across the world

Explore the crypto world
Buy and sell cryptocurrencies
easily on DirectX



Latest Transactions

From: 0x165...3320
To: 0xB57...e28E
Amount: 0.02 ETH

Message: sky

4/27/2022, 12:13:36 PM

From: 0x165...3320
To: 0xB57...e28E
Amount: 0.01 ETH

Message: Fast

4/27/2022, 11:06:08 AM

From: 0xB57...e28E
To: 0x165...3320
Amount: 0.01 ETH

Message: speed

4/26/2022, 8:45:42 AM

From: 0xA5a...9bC2
To: 0xbBC...B06b
Amount: 0.001 ETH

Message: cr7

From: 0xB57...e28E
To: 0x165...3320
Amount: 0.2 ETH

Message: nice

From: 0xB57...e28E
To: 0x165...3320
Amount: 0.01 ETH

Message: fire