CLP Token

Requirements and Potential Approaches

Furqan Ahmed

2019

Contents

[Problem Statement: 1](#_Toc12224550)

[Initial understanding and Solution (and why it is not feasible): 1](#_Toc12224551)

[First Approach: 1](#_Toc12224552)

[Second Approach (~**Blockchain interoperability)**: 2](#_Toc12224553)

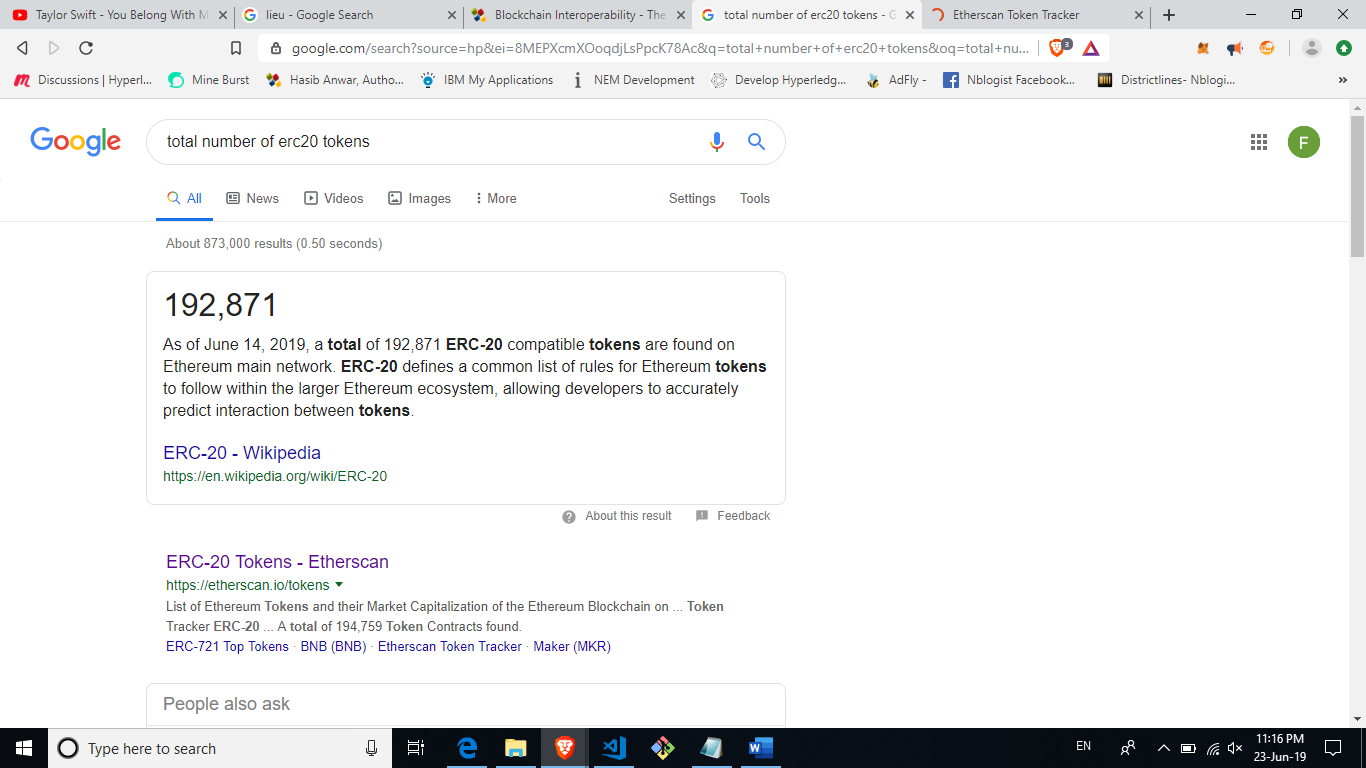
# 

# Problem Statement:

A user should be able to convert whatever currency it has to CLP Tokens (Defined as ERC20 Standard Token on Ethereum – Test Token at [0xbd907aaa9061ac55604c5e9abae7d77ef6abf225](https://rinkeby.etherscan.io/token/0xbd907aaa9061ac55604c5e9abae7d77ef6abf225)) There are two approaches through which this can be reached.

# Initial understanding and Solution (and why it is not feasible):

Initial understanding of the author was that a user would send any ERC20 Token to the contract and in exchange the Contract will send CLP Tokens with help of getting the exchange rates on different markets like CMC, CC etc.

The initial approach to this problem was to make an ERC20 CLP Token and another contract to deal with the exchanges, upon working on the exchanges module in the contract, the author discovered it wasn’t possible first because the contract need to have been programmed to have wallets for all these different ERC20 Tokens else all the tokens sent would be burnt (Lost forever), second part that makes this approach infeasible is :  and more coins get added daily, just like we’ll be adding our CLP Token on Main Net.

This lead to the better understanding of the author about the requirements, which were clarified on 23th June 2019.

# First Approach:

In this approach we won’t be able to hold all the different Tokens/Coins with us, rather will only get ether as payment and any unsold CLP Tokens.

Steps

1. Deploy an ERC20 Token – Completed successfully on testnet with a total supply of 1m and 4 decimals (Test Token at [0xbd907aaa9061ac55604c5e9abae7d77ef6abf225](https://rinkeby.etherscan.io/token/0xbd907aaa9061ac55604c5e9abae7d77ef6abf225))
2. Deploy another smart contract to deal with all the sale of CLP Tokens – Incomplete, basic functions have been defined, including the one to flush the account once tokens have been sold.
3. Make use of Oracles to connect to a crypto Exchange like shapeshift.io (Uses KYC now, not recommended to use) or Flyp.me (Recommended)
4. Next, when the user comes to our front end it would be given an option to select the coin it has, and the Ethereum address on which it wants to receive the CLP Tokens, upon which the API will generate an address of that Currency on which user will send funds, once received they’ll be sent to the contract as ether; Upon which the contract would send CLP Tokens (According to current exchange rate (CMC and CLP Token Business Logic), it could either be between the primary currency (BTC etc.) vs dollar -> CLP Token *or* Ether -> CLP Token) to the user’s Ethereum address. – Needs extensive use of off-chain Communication, will require use of Oracles (E.g. ChainLink) and some research on Flyp.me API is needed to know what functionality it provides – as ShapeShift.io API is out of question given recent KYC.
5. This way our Contract would have sold CLP Tokens in lieu of ether.

Note: We won’t be able to hold the different currency the user will send, we’ll only get ethers (Current Rate: 1ETH = $330).

# Second Approach (~**Blockchain interoperability)**:

This approach makes an *educated guess* for keeping in view the requirement of Mr. Shahmeer for being able to hold all the different currencies a user might send.

What we are trying to do here resembles a lot to Blockchain interoperability projects in development, at least to me. Here we try to use a different approach to try and solve the problem at hand

**Blockchain interoperability:** Is a concept under development that lets different blockchains “talk”, enabling cross chain transactions.

Steps:

A user sends a currency to some multi-currency wallet (Say, Coinbase) (again will need APIs for that) when the account detects money at one of the currencies on which the user have sent funds, using an oracle, it should communicate with our token sale contract to send relevant amount of CLP Token (With CLP Token Business Logic given in Token Economies Document) to the user’s Ethereum address, This way we’ll be able to hold all of the currencies that a user sends given that Coinbase multi-currency wallet support that currency. – Need for Extensive use of oracles, and Research into CoinBase API.

Look into UNISWAP.