Asset Tokenization

Requirements and Design Document

Estella Yeung, 100316780

April 16,2020

Introduction

Advancement in blockchain technology has provided the foundation for a revolutionized approach to asset ownership - the fractional ownership concept.

Fractionalized real estate ownership is not here yet due to legal and regulatory requirements. If it has happened, fractional real estate owners should be aware that their fractional ownerships are not protected under the existing laws and regulations in Ontario.

This asset tokenization project comes out of my interest in fractional ownership through blockchain technology. I believe such change and acceptance could revolutionize the industry worldwide bringing financial benefits to both owners and investors.

Asset tokenization can be on any asset class. The focus of this project is on real estate, specifically, the homeowners.

I do not have the technical knowledge to fully implement this project but would like to use this opportunity to examine the process flow and areas of concern, and use this project as a feasibility study, or proof of concept, or a learning experience.

Assumptions:

- The property has no mortgage
- The property does not receive rental income
- The property is the homeowner's principal residence, not a rental property

What is Asset Tokenization

- It is the process of creating fractional ownership on an asset through blockchain-based tokens
- It helps to make the illiquid real estate assets, liquid
- It is similar to reverse mortgage but much easier to execute
- It helps homeowners to raise capital easily with much less expense

How Asset Tokenization Benefits

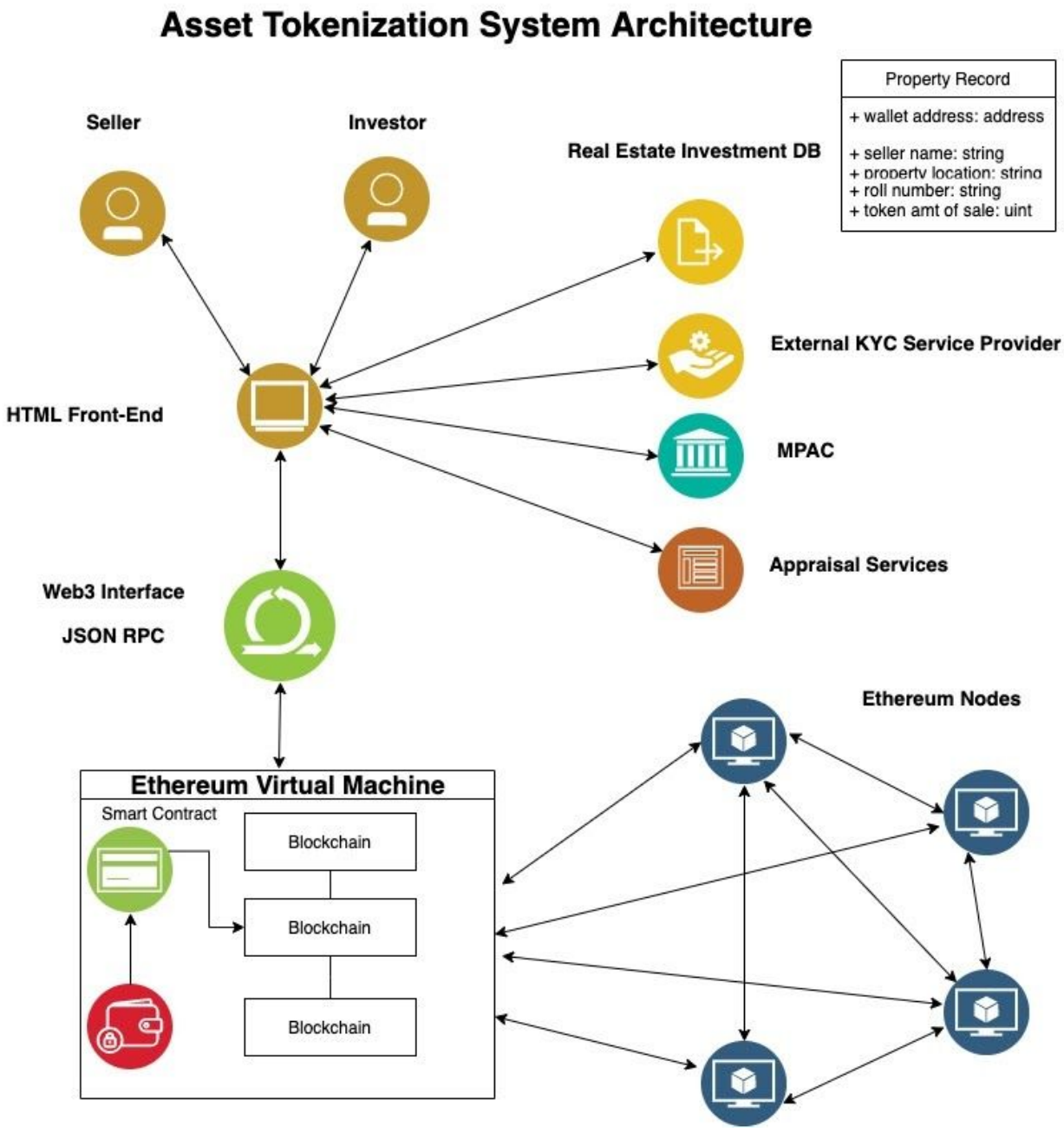
- Improve asset liquidity: investments in real estate have traditionally been illiquid because the owner either owns the whole property or co-owns the property. Owning a fraction of the property is not allowed in the existing laws and regulations of Ontario
- Faster settlement: smart contracts have logics to handle automated token transfer and settlement. As soon as the transaction is complete, the ownership record is stored on the blockchain and it is permanent and immutable
- Much cheaper to raise money on personal property
- Tokenization benefits both property owners and investors

Features that Support

In this asset tokenization project:

- Users will have an easy to use front-end App
- The system is linked to Municipal Property Assessment Corporation for the latest assessment value on the property
- The system is linked to Appraisal Services for the latest assessment value performed by qualified professionals
- The system is linked to KYC service provider to validate user identity, which also helps to prevent money laundry and counter-terrorist financing
- The system has an upper limit of not more than 40% of the property value; user to determine the percentage tokenization within this limit
- The value exchange is on ETH, i.e., ether

Asset Tokenization System Architecture



Asset Tokenization System Design

HTML User Interface

User interface is the most important part of the system because it is here the system prevents overloading the blockchain and it:

- Verifies the user to comply with regulatory requirements
- Verifies the real estate asset against local real estate registration system and in Toronto, it is the MPAC (municipal property assessment corporation)
- Verifies the latest property appraisal to ensure the value is fair to protect investors
- Calculates a harmonized property value
- Limits the fractionalized percentage on the property to protect investors
- Allows investors to browse the real estate investment server where it contains properties that are already fractionalized and ready for the investor's consideration

Another important job the user interface does is to allow the seller to register tokenized property in the real estate investment data store for the investor to browse investment opportunities. This is necessary because chances for the seller and the investor to be in the system at the same time completing the transaction is most likely not happening.

- User logon
    - User should have already obtained an ETH wallet address
    - Require for both seller and investor
        - Name
        - Home Address
        - ETH wallet address
        - Security Id. In Canada, it is SIN number; outside Canada, it is the valid passport number
    - User security check through an external KYC service provider to validate identity, to prevent anti-money laundering, for counter terrorist financing compliance, and for any other red flags
        - If not pass user validation, stop processing and exit system
    - If seller, user to enter
        - Property address
        - Property Assessment Roll Number
        - Desired fraction amount to sell

- If investor, user should have already browsed the Real Estate Investment Opportunities server and to enter
  - Property address
  - Property Assessment Roll Number
  - Desired fraction amount to purchase
- System to validate input data against MPAC database to retrieve
  - MPAC assessment on Assessment Roll Number
  - If not found, print a message ‘Property not found.’ End processing
- System to validate input data against the Appraisal Services database to retrieve
  - The latest appraised value and assessment date on Assessment Roll Number (to let the investor know when appraisal was performed)
  - If not found, print a message ‘Seller is required to provide the latest appraisal on the property.’ End processing

    Note: the check is to ensure the seller has the appraisal done

- System to calculate the harmonized value
  - Add the MPAC assessment, the appraised value and divide by 2
  - Calculate if the seller desired selling fraction is within asset value range of 40%. If over, generate a message ‘The desired selling amount is over asset value range of 40%.’ End processing
- For seller, system to write a record to the Real Estate Investment DB the following:
  - Seller ETH wallet address
  - Seller name
  - Property location
  - Assessment roll number
  - Token amount of sale
- For investor, system to send to the blockchain via Web3 interface the following:
  - Investor ETH wallet address
  - Seller ETH wallet address
  - Property location
  - Assessment roll number
  - Token amount of purchase

Note:

- Appraisal is required to ensure the latest property value and to prevent over-valuation

Connection to the Ethereum Blockchain

From the user interface, connection to the Ethereum blockchain is via Web3.js interface. From here, it changes the application into the blockchain application.

The Web3 JavaScript library is a collection of libraries which allow interaction with the local or remote Ethereum node through JSON RPC using HTTP or IPC connection. It can retrieve user accounts, send transactions, send Ether from one account to another, interact with smart contracts, read and write data from smart contracts, and create smart contracts.

Example of Web3 use.

Account:

- web3.eth.getAccounts : get account from Ethereum node
- web3.eth.getBalance : get account's balance
- web3.utils.fromWei : convert wei to ether

Transfer Ether from "A" to "B":

- txnObject (transaction object) is first argument of web3.eth.sendTransaction. txnObject is made of JSON.
- from : String -The address for the sending account.
- to : String — the destination wallet address.
- value: Number|String|BigNumber — (optional) the amount of Ether you wish to send to the destination address.

Ethereum Smart Contracts

ERC20Token.sol and AssetTokenizationContract.sol

- Based on ERC20 token standard
- Use safe math
- Need to keep track of balances
- Each token represents a share of the unique real estate
- When making a purchase:
    - Need to transfer ether from one account to the other. This means pay money from the investor to the seller account
    - Send ether to the seller wallet
    - Emit a purchase event with the investor's address and the amount
- Update the off-chain Real Estate Investment DB:
    - Calculate the remainder token amount = for sale amount - purchased amount
    - Update the remainder token amount of the property record using the following match:
        - Seller ETH wallet address
        - Property location
        - Assessment roll number
        - Remainder token amount of sale

Escrow.sol

- Should have a third party to oversee the real estate transaction - the agent
- Use the agent to deploy the contract
- This third party agent to deposit money into the escrow account
- This third party agent to withdraw money from the escrow account and deposit the fund to the seller, who is the payee
- Only the escrow agent can do the withdraw and deposit work to prevent issues

TokenRelease.sol

- Beneficiary is the owner of the property, i.e., the seller
- Hold tokens until release time
- Require system block time greater than or equal to time lock stamp
- Require beneficiary address
- Require token amount is greater than zero
- When that happens, transfer the tokens to the beneficiary address

Asset Tokenization Issues

- Real estate ownership partial transfer through blockchain is not recognized in existing Ontario law
- If the homeowner wants to tokenize the property and the property has a mortgage, this issue must be resolved with the mortgage holder first before going through tokenization. The situation could turn nasty and messy if the mortgage issue is not resolved first
- If part of the property generates income, it should be disclosed and how income from the property could be divided. If income is not disclosed, it could be an issue
- If the property involves a dispute, how claims right is resolved in a tokenized asset? The existing law may not have an answer. Investor beware

Project Roadmap

High Effort Medium Effort Low Effort	Year 1			
Initiative - Impact H, M, L	April	May	June	July
Requirements and Design Document (H)	---------			
Governance and Risk Assessment (H)	---------	--------		
Regulatory Compliance and Risk Management Strategy (H)		--------	--------	--------
Project Funding Drive (H)	---------			
Develop Activities to Identify the Occurrence of a Security Event		--------	--------	
User Interface DDI, Web3 Interface, Off-Chain Data Store and Retrieve Testing (H)		--------	---	
Smart Contract DDI, and Testing			----	---
End-to-End Testing				-------
Network and Data Security Audit			--------	--------
Program and System Documentation				--------
Emerging Trends	---------	--------	--------	--------

Project Development and Test Environment

- ganache
    - A personal blockchain for Ethereum development used to deploy contracts, develop applications, and run tests
- truffle
    - Get a boilerplate from truffle to speed up development
    - Compile contracts, deploy contracts, inject it into a web app, create front-end for dApps and testing
    - v5.1.18 is the existing truffle version in the system
    - From truffle, can do the following
        - truffle compile
        - truffle console (to interact with web3
- Mocha allows asynchronous testing, test coverage reports, and use of any assertion library
- Chai is a BDD / TDD (behaviour driven development / test driven development) assertion library for NodeJS and the browser
- Metamask for google chrome
    - Allows connection to the blockchain and interacts with the smart contract
- npm install

Test blockchain connection

```
truffle(development)> accounts = await web3.eth.getAccounts()
undefined
truffle(development)> accounts
[ '0xD27ACA204D6fCbaE6D9538F93632fFC260299612',
  '0xd473722ff3216bD7933e39d19f6B9A2551f2b013',
  '0x28A55FaC2bdFb603173B68a3550e10d5e706B33D',
  '0x1e538b30894D119b5Bfb8c9A3213016533E3f82A',
  '0x698ef4Fc9Dc985059D6d99cBe25f8A10F812A321',
  '0x1fA7E792f1CFf5CA14A885B4CA6b77d96d460740',
  '0x41e053877E3e3507616443382840B3B46c08EBF4',
  '0xdE83a1dB62E6FAe0D3Ed658CCbB98b3F3CBeE461',
  '0xEE0D36F48B3933F2a10f8071c24b8A9Fbe703a55',
  '0xC82B599aB56D1b263b34581B6317e3261a737B5B' ]
truffle(development)>
```

```
truffle(development)> blockNumber = await web3.eth.getBlockNumber()
undefined
truffle(development)> blockNumber
4
truffle(development)>
```