Software Requirements Specification (SRS) Document

Own Your Home
Team 5

Naval Surange | Siddharth Jain | Akash C.R. | Divyansh Tiwari

Brief problem statement

The project asks us to develop a low code/no code Web 3.0 based app/protocol that allows users to fractionalize their house ownership by establishing interoperability of digital NFTs of sections of their property in exchange for cryptocurrency/ digital tokens, allowing them to be free of the high interest debt created by the mortgage providers and banks

System requirements

The final goal of this project is to create a web 3.0 based website and a custom contract through which users can create digital NFTs of their real-world assets like a house or Real estate and then fractionalize this newly created NFT.

The Project should have the following features,

1) The website should be able to connect to digital wallets present on your PC like Meta-Mask wallet, etc.

2) We should be able to see the assets and liquidity available from other DAOs (like Centrifuge/Tinlake, CitaDao, Synthetix) and trade them in our platform.

3) The contract should be based on the ERC721 or ERC1155 (https://moralis.io/erc1155-exploring-the-erc-1155-token-standard/) token.

4) The front end could be implemented in Word-press or Bubble.

5) The smart contract would be built in Ethereum L1 for consensus, Polygon (or zk-rollup/sync) for Data and Execution.

Users profile

The project would be primarily used by the following users:

1. Asset Originator - These will be the (property owners) who would use our website to convert the property into a digital asset and fractionalize them.
   a. Familiarity with using computers – low to medium
   b. Familiarity in using a similar website – low to expert
2. The speculators – They provide the capital that the people are betting on. They will buy/sell fractions of NFTs
   a. Familiarity with using computers – low to medium
   b. Familiarity in using a similar website – low to medium
3. Community Managers (earn a referral bonus, might also be mortgage brokers) –
   a. Familiarity with using computers – low to medium
   b. Familiarity in using a similar website – intermediate to experts
4. Mortgage Providers (NBFC+banks) : Token/NFT Buyers – These will be the people/companies who would buy the digital assets in exchange for some rights over the fractionalized property.
   a. Familiarity with using computers – low to medium
   b. Familiarity in using a similar website – intermediate to experts

- To mortgage brokers it provides another option of directing finances to homeowners,
- To mortgage providers and banks, it provides further liquidity in the mortgage market,
- To homeowners, it provides a way to raise finances against their equity in the property,
- For Bridge loan and mezzanine financiers, there is a market that they can tap to help owners of properties go 'on-chain'

Use case diagram

Read the instructions below and fill in the table. Delete all the blue text before adding this to your repository or turning it in to your instructor.

Draw the UML use case diagram for the system. Make sure the use cases shown in the diagram correspond to the use cases described in the previous section.

Feature requirements (described using use cases)

No.	User Case Name	Description	Release
1.	*Adding Property*	*The asset originators/providers should be able to convert their real assets to on-chain digital NFTs. (by applying, then getting the asset approved by community managers (muti signature), getting the Special Purpose Vehicle (SPV set up in the form of a Limited Liability Company LLC), and then listing the asset/NFT on the platform for a 'party' bid, or group bid from speculators and mortgage providers to bid on)*	*R1*
2.	*Token for exchange*	*There should be a token based on the ERC721/ERC1155 standard through which people can exchange digital NFTs*	*R1*
3.	*Fractionalization of the NFT*	*The asset originators/providers should be able to fractionalize the digital NFT created. (eg. https://www.partybid.app or https://fractional.art/)*	*R1*
4.	*Adding property rights*	*asset originators/providers should be able to specify what rights over the property (rights transfer language to be vetted by Mortgage brokers/community managers during the asset evaluation) a speculator will get when they buy a fractionalized part.*	*R1*
5.	*Authentication with user wallets*	*Any user visiting should be able to log in through the Web3 Ethereum wallets present on their systems like Meta-Mask etc.*	*R1*
6.	*Support for other Ethereum based tokens*	*Users should be able to use Ethereum or the community token of OwnYourHome DAO, or XDC https://xinfin.org/ https://youtu.be/K-tHZkV6zAs https://tinlake.centrifuge.io*	*R2*

7.	*Interoperability with other DAOs*	*Buyers should be able to see the NFTs available on similar webapps and should be able to trade them through our web3 platform (https://github.com/ethereum-boilerplate/ethereum-nft-marketplace-template)* https://github.com/Hrushi2406/nft-marketplace https://tinlake.centrifuge.io https://opensea.io/	*R1*
8.	*Extended Interoperability*	*Our NFTs should be shown on other DAOs' websites as well and users there should be able to buy them. (backwards compatibility bridge)* https://tinlake.centrifuge.io https://governance.aave.com/t/add-fireblocks-as-a-whitelister-on-aave-arc/5753	*R3*
9.	*Verification of property.*	*The verifier should be able to submit proof that the asset originators are the real owner of the property (this typically involves the full transfer of the property into the SPV LLC (special purpose vehicle LLC)*	*R1*
10		https://rupya.finance/ https://gamla.fund/	

Use case description

Use Case Number:	UC-XX (Replace XX with a number)
Use Case Name:	Enter the name of Use Case

Overview:	Describe the purpose of the Use Case and give a 1-2 line description. This could be the same as the description provided in feature requirements section.
Actors:	List all actors that participate in this Use Case.
Precondition:	Enter the condition that must be true before the main flow is executed.
Flow:	Main (success) Flow: Steps should be numbered.
	Alternate Flows: Include the post condition for each alternate flow if different from the main flow.
Post Condition:	Enter the condition that must be true when the main flow is completed.

Test Backlog

No backlogs as of yet.