

Software Design

for

Financial Application

A Decentralized Exchange

FunixSwap

Version 1.0

**SIGNATURE**

**ORIGINATOR: Pham Tien Dung – SE63086**

**REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

**Contents**

[1. Product overview 4](#_Toc22763118)

[2. Functional Specifications 5](#_Toc22763119)

[2.1. Use Cases Diagram 5](#_Toc22763120)

[2.2. Use Cases Specifications 6](#_Toc22763121)

[a. Swapping tokens 6](#_Toc22763122)

[b. Transfering tokens 7](#_Toc22763123)

[c. Wallet Information 8](#_Toc22763124)

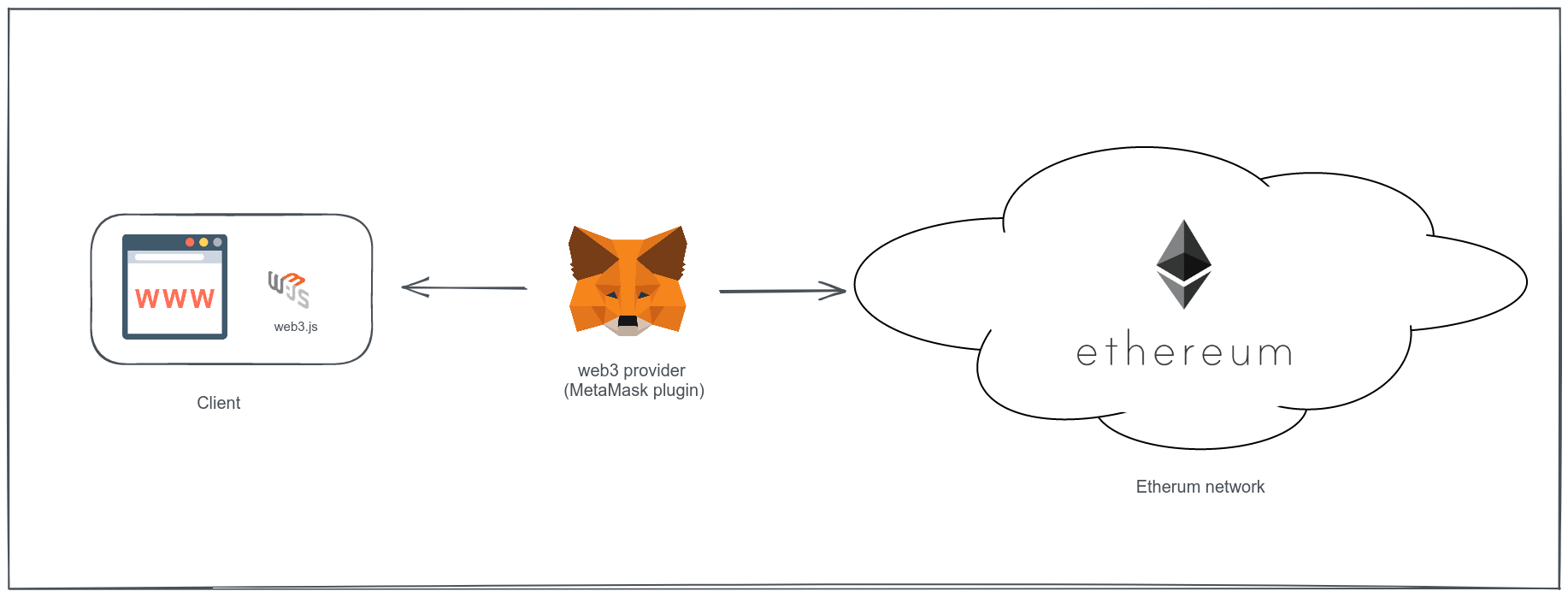
[d. Importing Wallet 9](#_Toc22763125)

[e. Transaction Monitoring 10](#_Toc22763126)

[3. Non-Functional Requirements 11](#_Toc22763127)

[4. Product Upgrades 11](#_Toc22763128)

# 1. Architecture



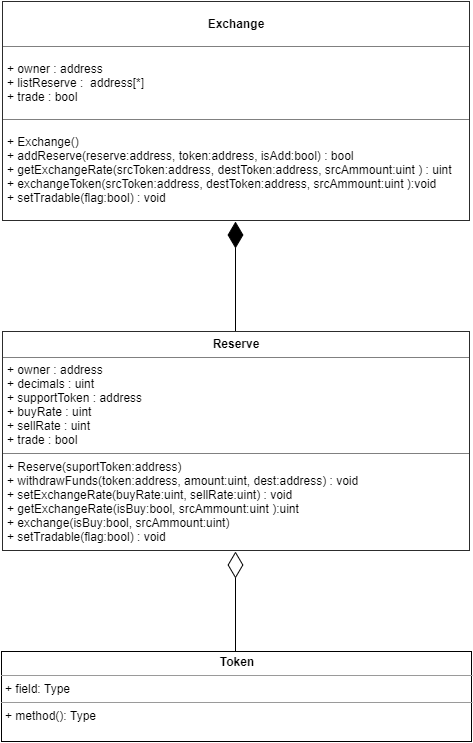
My project will be based on NodeJS with important library is Web3Js

“web3.js is a collection of libraries which allow you to interact with a local or remote ethereum node, using a HTTP or IPC connection.” - web3js.readthedocs.io

And Metamask extension is used as a bridge between client and Ethereum Network.

In Ethereum network, client only interact with Transfer contract. As our requirement, there are at least 2 custom tokens must be deployed, and they are controlled by 2 Reserve smart contract, which are managed by Transfer smart contract.

# 2. Class Design



# 3. Functional Design

1. Reserve contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Function name** | **Function purpose** | **List of parameters** | **Return type** |
| 1.1 | Reserve | Constructor | supportToken:address | void |
| 1.2 | withdraw | Allow owner withdraw token | token: address  amount: uint  dest: address | void |
| 1.3 | getExchangeRate | Get exchange rate of token and Eth | isBuy: bool  amount: uint | uint |
| 1.4 | setExchangeRate | Set exchange rate of token | buyRate: uint  sellRate: uint | void |
| 1.5 | setTradeFlag | Set value trade flag (allow trading or not) | flag: bool | void |
| 1.6 | exchange | Exchange token with reserve contract | isBuy: bool  amount: uint | void |

1. Exchange contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Function name** | **Function purpose** | **List of parameters** | **Return type** |
| 2.1 | Exchange | Constructor | N/A | void |
| 2.2 | addReserve | Add or remove a reserve | reserveAddress: address  tokenAddress: address  isAdd: bool | bool |
| 2.3 | getExchangeRate | Get exchange rate of 2 tokens | srcToken: address  destToken: address  amount: uint | uint |
| 2.4 | exchangeTokens | Exchange between 2 tokens | srcToken: address destToken: address  amount: uint | void |