

```

??Contract Octa-tron.sol document??

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.3;

/**
 * @notice octa-tron solidity contract.
 * @dev User specifies exact input (msg.value).
 */

contract Ownable {

    /**
     * @dev The Ownable constructor sets the original `owner` of the contract to
the sender
     * account.

     * @dev Throws if called by any account other than the owner.

     * @dev Allows the current owner to transfer control of the contract to a new
Owner.
     * @param _newOwner The address to transfer ownership to.
     */
}
constructor() public {
    /**
     * @dev Constructor Sets the original roles of the contract
     */
}

function payable() external payable {

//default to buy plan 0, no referrer

}

function _init() private {
    investmentPlans_.push(Objects.Plan(27,79*60*60*24,37)); //79 days
    investmentPlans_.push(Objects.Plan(37, 48*60*60*24,47)); //48 days
    investmentPlans_.push(Objects.Plan(47, 28*60*60*24,57)); //25 days

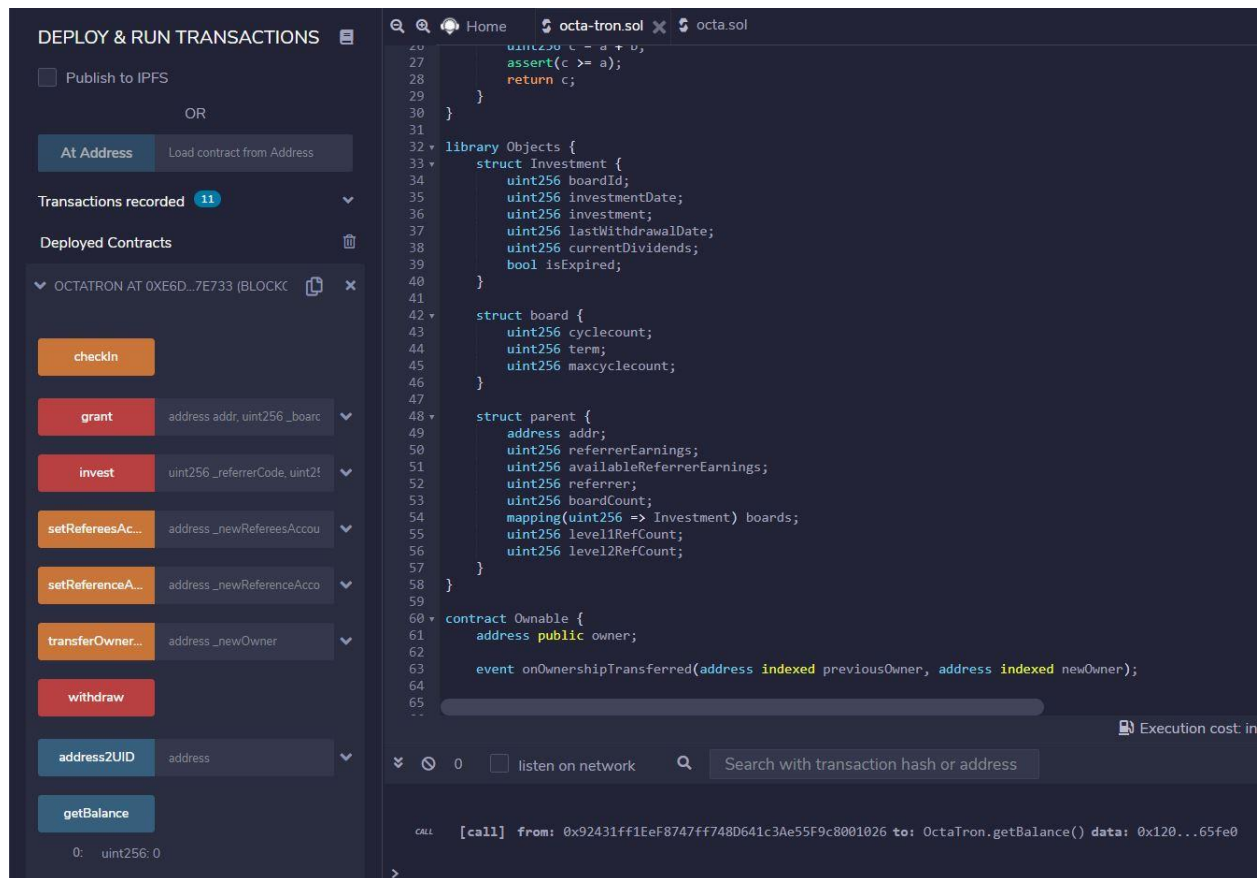
```

```
        investmentPlans_.push(Objects.Plan(57, 20*60*60*24,67)); //18 days
    }

function _addreferee(address _addr, uint256 _referrerCode) private returns (uint256) {
    //require(uid2Investor[_referrerCode].addr != address(0), "Wrong referrer code");

}

function _invest(address _addr, uint256 _planId, uint256 _referrerCode, uint256 _amount) private returns (bool) {
    //new user
} else { //old user
    //do nothing, referrer is permanent
}
```



Checkin:

Requirement:

1. No need to register with email and phone number

Solution:

User directly joins the network by a link (for only one time) only metamask wallet will connect upon checkin.

Grant:

Requirement:

3. Everyone starts with the \$25 board, and then the \$100 and then \$300 follow after a user has referred two other individuals (wallets)

Solution:

System/admin generates the required bonus/grant upon individual referred others

Referrer code

Requirement:

2. When a person joins, they use a link that will be provided by the person who invited them.

Solution:

Link or code for a person who join the network

Owner/ getReferenceAccount/ getRefereesAccount /
getDeveloperAccount

Requirement:

4. Each person in the platform will be identified by number (*No names*) the number forms part of the link so the system knows who has invite who.

Solution:

A public key/ address of the user in the network

Time-stamp implemented in the contract

Requirement:

7. As people join, the Smart Contract shall time stamp every person so that they are allocated the positions on the board in a time-stamped fashion.
8. Each person's referrals will follow them in whatever board they go to, and if one board has been filled but the members of that board keep inviting people, there will be a spillover allocated in a time fashioned way.

Solution:

System realizes FIFO (First come First serve) request.

Withdraw

Requirement:

7. Payouts are paid directly to the individual wallet, and the person can use that payment however they want.

Solution:

Tokens will be added in the available wallet. Currently in metamask.

Reference-level1-rate + Reference-level2-rate

Requirement:

12. Referral link when given to a minimum of two referees, and they join OCTA-TRON, this qualifies the referrer to move from the ALPHA Board to the on to the OMEGA Board, "Parent" and the referred person "Child" can follow the parent automatically as long as they too bring in their 2 people (referrals)

Solution:

2 referee/referral link per user

Time Stamp

Requirement:

13. If after the first cycle you have not referred anyone, the smart contract would pay you and you automatically exit at that initial stage. Should you want to continue, you may have to pay another \$25 and start afresh. But if you have already referred your two, The Smart Contract will automatically pay you the residual and that the next amount to be invested in Cycle 2, 3 and so fourth until the last cycle where everyone after they are paid they will start all over at \$25 making this system unique in the sense that everyone gets paid the same. No one gets paid more than another.

Solution:

System remove users in a time fashioned manner, currently number of days are random i.e. 27, 47 and must be adjusted before deployment according to the requirement

Fall back function implemented

Requirement:

16. The Platform fee would be used for administrative purposes and build up an insurance fund for any eventualities that may arise unexpectedly to cover any losses that the community members may have suffered to at least reimburse those who may still be caught in the system before they are paid anything.

Solution:

Receiving end of the server following the contract requirement