BLOCKCHAIN and REAL ESTATE

A GAME-CHANGER

Team 4

Smart Contracts for Real Estate Management

NAMES OF OUR PRESENTERS:

- YUJIE CHEN
- SHWETA UPADHYAY
- PARASTOO AZIMI

Objective:

Demonstrating the power of smart contracts in real estate

Team 4

Agenda:

- Introduction to Smart Contracts
- Property Verification Smart Contract
- Real Estate Tokenization Smart Contract
- Real Estate Marketplace Smart Contract

Team 4

SMART CONTRACTS:

- Self-executing contracts with the terms directly written into code
- Automatically enforce, facilitate, or verify agreements
- Execute on blockchain platforms like Ethereum
- Eliminate the need for intermediaries

Team 4

VERIFICATION SMART CONTRACT

- Contract Overview:
  - Manages property verification
  - Allows adding and verifying properties
  - Ensures ownership and existence of properties

Lear More

100+

More Property

The Property Verification Smart Contract serves as the guardian of property authenticity on the blockchain. It allows property owners to add and verify their assets, ensuring ownership and existence. This contract creates a trustless and secure environment for real estate transactions, revolutionizing property management.

```solidity
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract PropertyVerification {
    struct Property {
        address owner;
        string HouseDetails;  // IPFS or other storage hash
        bool exists;
    }

    mapping(uint256 => Property) public properties;
    uint256 public propertyCount = 0;
    address public owner;

    constructor() {    686961 gas 657000 gas
        owner = msg.sender;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    event PropertyAdded(uint256 indexed propertyId, address indexed owner, string houseDetails, bool exists);
    event PropertyVerified(uint256 indexed propertyId);

    // Adds a property and returns its ID
    function addProperty(string memory houseDetails) public onlyOwner returns (uint256) {    infinite gas
        propertyCount++;
        properties[propertyCount] = Property({
            owner: msg.sender,
            HouseDetails: houseDetails,
            exists: true
        });

        emit PropertyAdded(propertyCount, msg.sender, houseDetails, true);

        return propertyCount;  // Returns the newly added property's ID
    }

    // Verifies a property and returns 1 for successful verification, 0 otherwise
    function verifyProperty(uint256 propertyId) public onlyOwner returns (uint256) {    infinite gas
    if (propertyId <= 0 || propertyId > propertyCount) {
        return 0;  // Invalid property ID
    }

    Property storage property = properties[propertyId];
    if (!property.exists) {
        return 0;  // Property does not exist
    }

    emit PropertyVerified(propertyId);

    return 1;  // Verification successful
```

Team 4

Property Verification contract code

- **Property struct:** This data structure holds information about a property, including the owner's address, property details (like a deed hash), and an existence status flag (true if verified).

- **addProperty function:** This function lets the contract owner add a property by providing property details. It increments a unique ID, creates a new property struct, and marks it as verified (exists = true). An event is emitted to confirm the addition.

- **verifyProperty function:** This function allows the owner to confirm the legitimacy of a property. It checks if the property ID is valid and if the property exists (i.e., it's been verified). If these conditions are met, it emits a verification event.

Simplification

Real estate is efficiently represented as blockchain tokens, simplifying the buying and selling of properties.

Transparency

Every property transfer is recorded, providing a transparent history of ownership changes for added trust and accountability.

Team 4

REAL ESTATE TOKENIZATION SMART CONTRACT

Contract Overview:

- Tokenizes real estate: This contract transforms real estate properties into unique blockchain tokens using the ERC721 standard.
- Stores details and history: It securely stores property information and transaction history.
- Enables transfers and records transactions: Property tokens can be bought, sold, and transferred, with every transaction meticulously recorded for transparency and accountability.

Team 4

REAL ESTATE TOKENIZATION SMART CONTRACT CODE

Explaining:

- **RealEstate struct:** This data structure encapsulates all property details, ensuring a comprehensive and organized storage mechanism for real estate information.
- **createRealEstateToken function:** This function creates new property tokens and registers them on the blockchain. It's exclusively accessible to the contract owner, ensuring control over property token creation.
- **transferWithRecord function:** This function enables the transfer of property tokens between addresses while simultaneously recording the transaction details. This records the history of property ownership changes, enhancing transparency and trust in property transactions on the blockchain.

```
            location,
            price,
            imageHash,
            propertyType,
            buildingType,
            storeys,
            landSize,
            propertyTaxes,
            parkingType
        );

        return newTokenId;
    }

    /**
     * @dev Function to view the details of a real estate token.
     * @return The detailed information of the real estate associated with the given token ID.
     */
    function viewRealEstate(uint256 tokenId) public view returns (RealEstate memory) {
        return realEstates[tokenId];
    }

    /**
     * @dev Internal function to record a transaction for a token.
     */
    function recordTransaction(uint256 tokenId, address buyer, uint256 price) internal {
        transactionHistory[tokenId].push(Transaction({
            buyer: buyer,
            price: price,
            timestamp: block.timestamp
        }));

        emit TransactionRecorded(tokenId, buyer, price, block.timestamp);
    }

    /**
     * @dev Function to retrieve the transaction history for a token.
     * @return The array of transactions associated with the given token ID.
     */
    function getTransactionHistory(uint256 tokenId) public view returns (Transaction[] memory) {
        return transactionHistory[tokenId];
    }

    /**
     * @dev Function to transfer a token and record the transaction.
     */
    function transferWithRecord(uint256 tokenId, address to, uint256 price) public {
        transferFrom(msg.sender, to, tokenId);
        recordTransaction(tokenId, to, price);
    }
```

Team 4

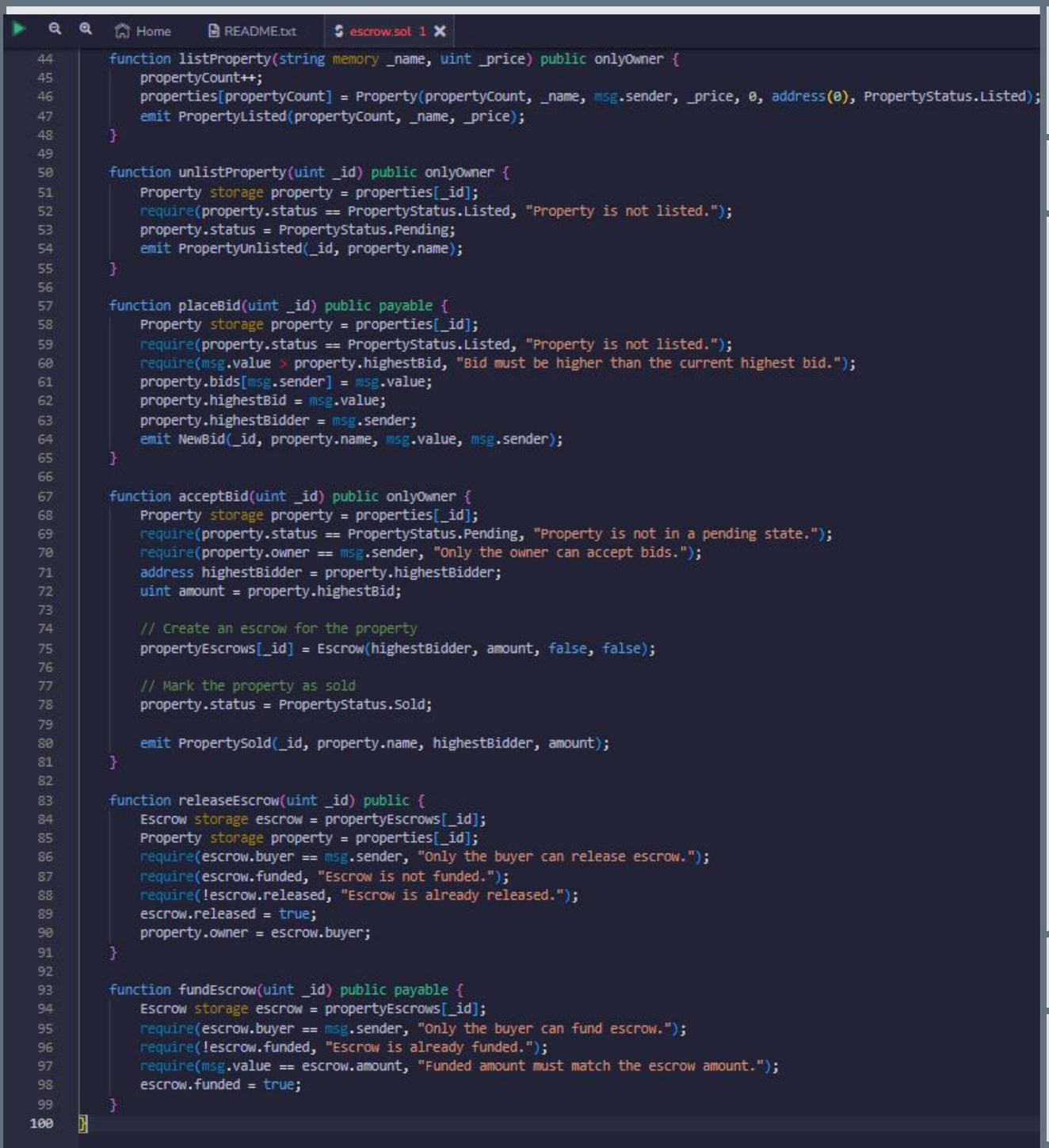
Escrow

In real estate, escrow is like a secure holding area and a middleman for important things like money and documents during the process of buying or selling a property.

The money is held by a neutral third party (the middleman) until all the necessary steps, like inspections and paperwork are completed.
Once everything is in order, the money is released to the seller

REAL ESTATE MARKETPLACE SMART CONTRACT

- Contract Overview:
  - Implements a real estate marketplace
  - Allows property listing, bidding, and escrow handling
  - Facilitates property sales and transactions



Team 4

REAL ESTATE MARKETPLACE SMART CONTRACT CODE

Real Estate Marketplace Smart Contract, manages the buying and selling of real estate properties. It includes the Property and Escrow data structures, functions like;

- **listProperty** for listing properties,
- **placeBid** for making offers, and
- **acceptBid** for finalizing sales.

The contract ensures a secure process for property transactions and the handling of **escrow** to protect buyers and sellers.

Best Seller In Town

Unlocking Real Estate's Digital Future: From Verification to Tokenization, Empowering Secure Transactions in Decentralized Marketplaces

Team 4

REAL-WORLD USE CASES

- Property verification before tokenization
Property verification ensures the authenticity of real estate before it is tokenized.

- Tokenization of real estate assets
Tokenization transforms real estate assets into digital tokens for easier management and trading.

- Creating decentralized real estate marketplaces
Decentralized real estate marketplaces allow peer-to-peer property transactions without intermediaries.

- Efficient and secure property transactions
Efficient and secure property transactions streamline the buying and selling of real estate with trust and transparency.

Team 4

BENEFITS OF USING SMART CONTRACTS

1. **Transparency:** Smart contracts provide a clear and open record of transactions and agreements.
2. **Trustlessness:** They eliminate the need for intermediaries, reducing the reliance on trusted third parties.
3. **Security:** Smart contracts use cryptography and decentralized networks to protect against fraud and tampering.
4. **Automation:** They automatically execute predefined actions when conditions are met, saving time and effort.
5. **Reduced Costs:** By cutting out intermediaries and automating processes, smart contracts lead to cost savings.


Team Four
THANK YOU
For sharing this spooky evening with us.
HAPPY HALLOWEEN