UNIVERSITY COLLEGE OF ENGINEERING, KANCHIPURAM

REAL ESTATE MANAGEMENT SYSTEM USING BLOCKCHAIN

Team Id: NM2023TMID00910

NAME	NM ID
V.R. NANDHINI	CEECC7353006016815D416196AA52476
R. PEREZHIL	CCC2B9481FE6308B48C4DDC591851E18
T.ARUNADEVI	7AB0D75F46B107FBE4345DA1966D2D8A
K. MAHALAKSHMI	E28AB86EF57E2E8A8EF44A81E101520C

REAL ESTATE MANAGEMENT SYSTEM

1. INTRODUCTION

1.1 PROJECT OVERVIEW

The Real Estate Management system using blockchain enhances

- ✓ Security
- ✓ Reduces the risk of fraud
- ✓ Including its potential to eliminate fraud
- ✓ Reduce intermediaries
- ✓ Increase transparency
- ✓ Make property transactions more accessible

Blockchain, often associated with cryptocurrencies like Bitcoin, is much more than just a digital currency. Whether you are a property owner, buyer, real estate agent, or investor, understanding the transformative power of blockchain in real estate is crucial for staying ahead in an ever-evolving industry.

1.2 PURPOSE

The Purpose of Blockchain in Real Estate Management System is,

- ✓ Enhanced Security and Immutable Records
- ✓ Elimination of Intermediaries
- ✓ Increased Transparency
- ✓ Efficient Property Title Management
- ✓ Fractional Ownership and Tokenization
- ✓ Smart Contracts for Automation
- ✓ Global Accessibility
- ✓ Historical Transaction Records
- ✓ Streamlined Property Valuation and Appraisals
- ✓ Compliance and Regulation Adherence

The purpose of implementing blockchain in real estate management is to bring about increased security, efficiency, transparency, and accessibility to the industry.

By leveraging the capabilities of blockchain technology, the real estate sector can overcome many of its existing challenges and provide a more streamlined and inclusive experience for all stakeholders involved

2. LITERTURE SURVEY

2.1 EXISTING PROBLEM

The real estate management system is associated with various challenges and problems, many of which have persisted over the years.

- ✓ Lack of Transparency
- ✓ Complex and Time-Consuming Transactions
- ✓ Fraud and Scams
- ✓ Inaccurate and Outdated Records
- ✓ High Transaction Costs
- ✓ Lack of Access for Small Investors
- ✓ Paperwork and Administrative Burden
- ✓ Disputes and Litigation
- ✓ Inefficient Property Management
- ✓ Lack of Standardization
- ✓ Regulatory Compliance Challenges
- ✓ Inadequate Data Sharing
- ✓ Limited Accountability

Addressing these problems is a significant challenge, but technology, particularly blockchain, has the potential to alleviate many of these issues by providing transparency, efficiency, and enhanced security to the real estate management system.

2.2 REFERENCES

- **Influence of Blockchain in the Real Estate Sector**
  M Nijland - International Journal of Applied Science, 2019 - j.ideasspread.org
  https://doi.org/10.30560/ijas.v2n2p22

- **Land ownership and land use development**
  Oliveira, E - the integration of past, present, and future in spatial planning and land management policies. Landsc. J. **36**(2), 119–121 (2017)
  https://doi.org/10.3368/lj.36.2.119

- **Legal challenges and opportunities of blockchain technology in the real estate sector**
  Garcia-Teruel, R.M. (2020), *Journal of Property, Planning and Environmental Law*, Vol. 12 No. 2, pp. 129-145
  https://doi.org/10.1108/JPPEL-07-2019-0039

- **Real Estate Land Transaction System Using Blockchain**
  Authors: Dipak D. Gaikwad, Akshay N. Hambir, Shantanu S. Chavan, Gayatri K. Khedkar, Dr. Shashikant V. Athawale
  https://doi.org/10.22214/ijraset.2022.40633

2.3 PROBLEM STATEMENT DEFINITION

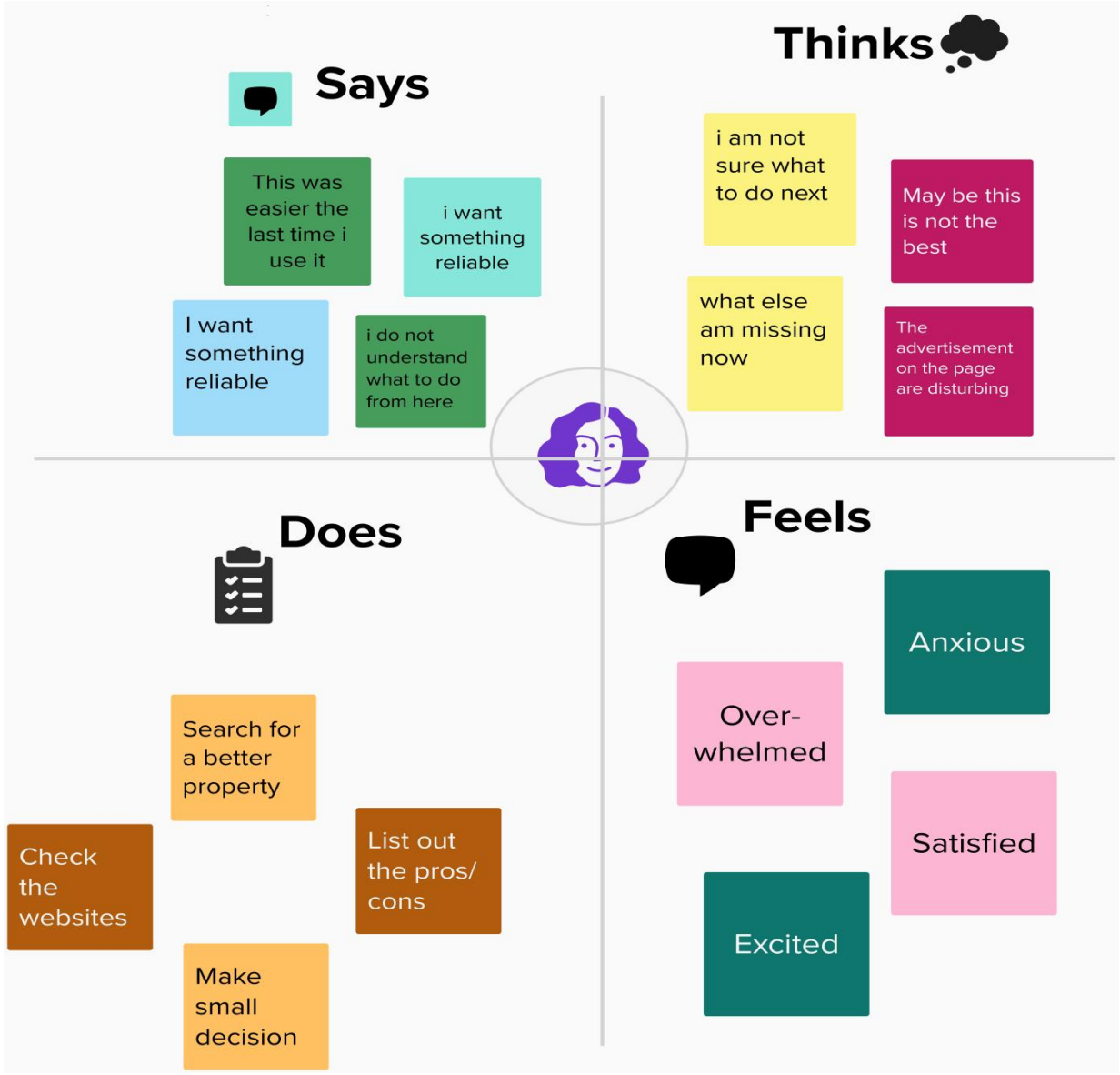
The problems that are encountered in Real Estate management system in real life are us follows:

- Data Accuracy and Integrity - Inaccurate property information or transaction data can lead to legal and financial issues. Maintaining data integrity and ensuring that information is up to date is critical.
- Security and Privacy - Protecting sensitive data, such as personal information and financial records, is paramount. Security breaches and data leaks can result in legal liabilities and damage to reputation.
- Scalability - Real estate management systems must accommodate growth as more properties and clients are added. Scaling up while maintaining performance can be challenging.
- User Experience - User-friendliness and accessibility for various stakeholders, including property owners, tenants, and agents, are crucial. A poor user experience can lead to decreased productivity and adoption.

- Maintenance and Updates - Regular maintenance, updates, and bug fixes are necessary to keep the system running smoothly.
- Cost Management - Managing the costs associated with maintaining and improving the system is essential. Budget constraints can limit the ability to address issues effectively.
- Customer Communication - Effective communication with clients and tenants is vital. Overlooking client inquiries and requests can lead to dissatisfaction.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION AND BRAINSTORMING

1. **Blockchain-Based Property Listings:**

Create a blockchain-based platform for property listings that ensures data accuracy, prevents fraudulent listings, and allows users to verify property ownership and transaction history securely.

2. **Tokenization of Real Estate Assets:**

Implement a system for tokenizing real estate assets, enabling fractional ownership and investment in properties. Users can buy and trade property tokens, reducing the barriers to entry for real estate investment.

3. **Smart Contracts for Transactions:**

Develop smart contracts that automate the entire property transaction process, from the initial offer to closing. This can include escrow services, digital signatures, and automated payment processing.

4. **Identity Verification:**

Utilize blockchain for secure identity verification and background checks for property buyers, sellers, and renters. Enhance trust and reduce the risk of fraud in real estate transactions.

5. **Property History and Records:**

Maintain a blockchain-based ledger of property history and records, including ownership changes, renovations, inspections, and maintenance. This ensures transparency and easy access to essential property information.

6. **Secured Property Data Storage:**

Create a decentralized and secure storage system for property-related documents, such as deeds, titles, and inspection reports. Property owners and authorities can access these documents easily while maintaining data security.

7. **Maintenance and Service Requests:**

Develop a platform for property owners, tenants, and service providers to manage maintenance and service requests using blockchain. Smart contracts can facilitate automatic scheduling and payment for services.

8. **Payment and Rent Collection:**

Implement blockchain-based payment systems for property rents, mortgages, and other financial transactions. This can ensure quick and secure payments with transparent transaction histories.

9. **Regulatory Compliance and Reporting:**

Create a feature that ensures compliance with local real estate regulations and automatically generates reports for property transactions, taxes, and other legal requirements.

10. **Data Analytics and Insights:**

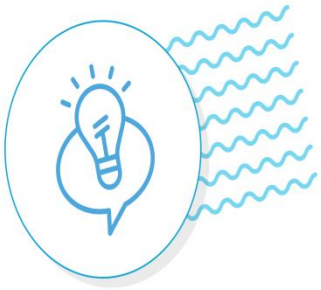
Provide data analytics and insights based on the blockchain ledger, helping property owners and investors make informed decisions about their real estate portfolios.

11. **Blockchain for Rental Agreements:**

Simplify the creation and management of rental agreements by using blockchain technology. This can include automated renewal notifications and rental payment tracking

3.3 IDEATION & BRAINSTORMING

PROBLEM STATEMENT



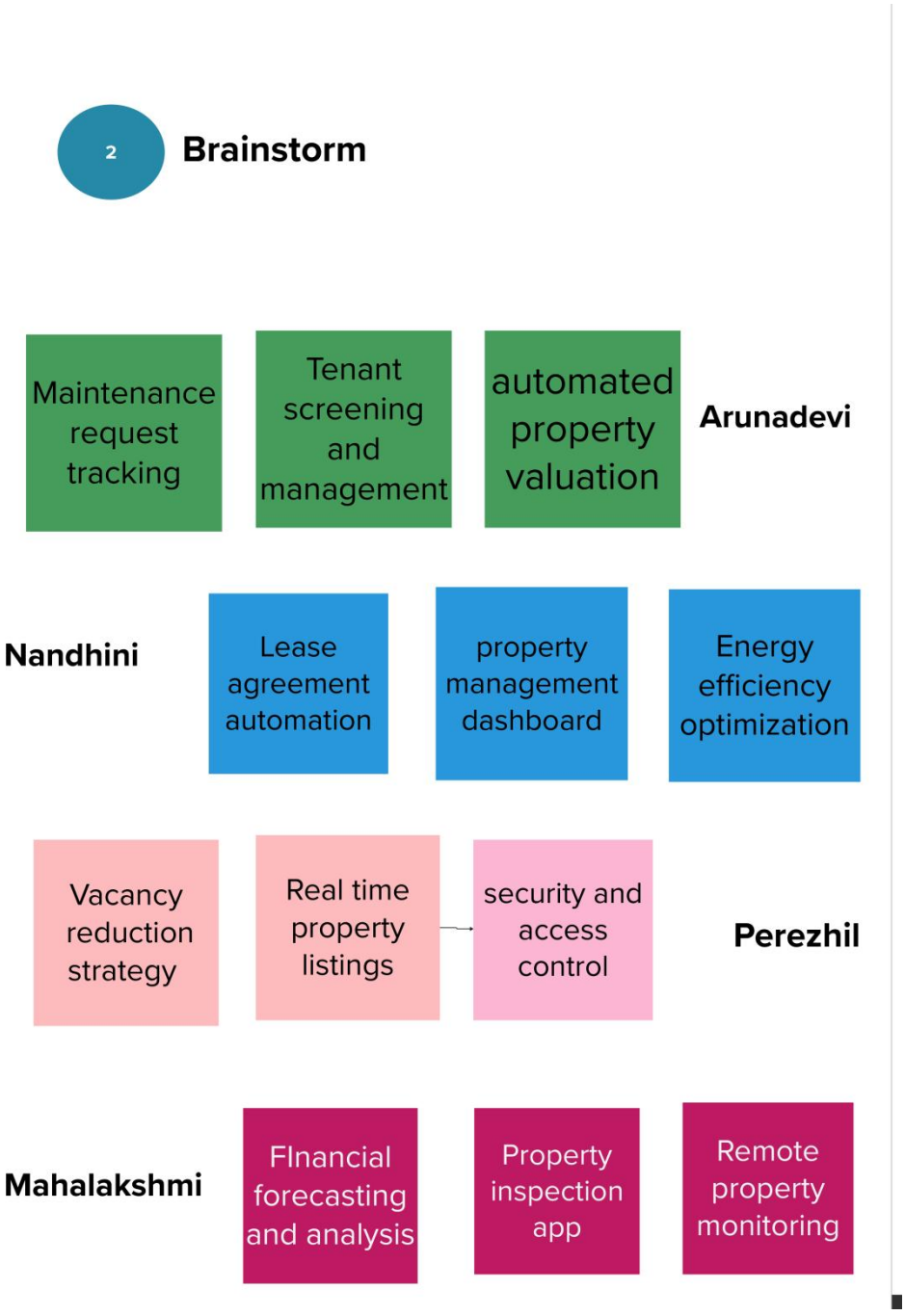
Brainstorm
& idea prioritization

1 **Define your problem statement**

PROBLEM

The current real estate management problems are often inefficient,time-consuming,lack transparency.Property owners,real estate agents,and tenants face challenges in property maintenance,rent collection,and communication.

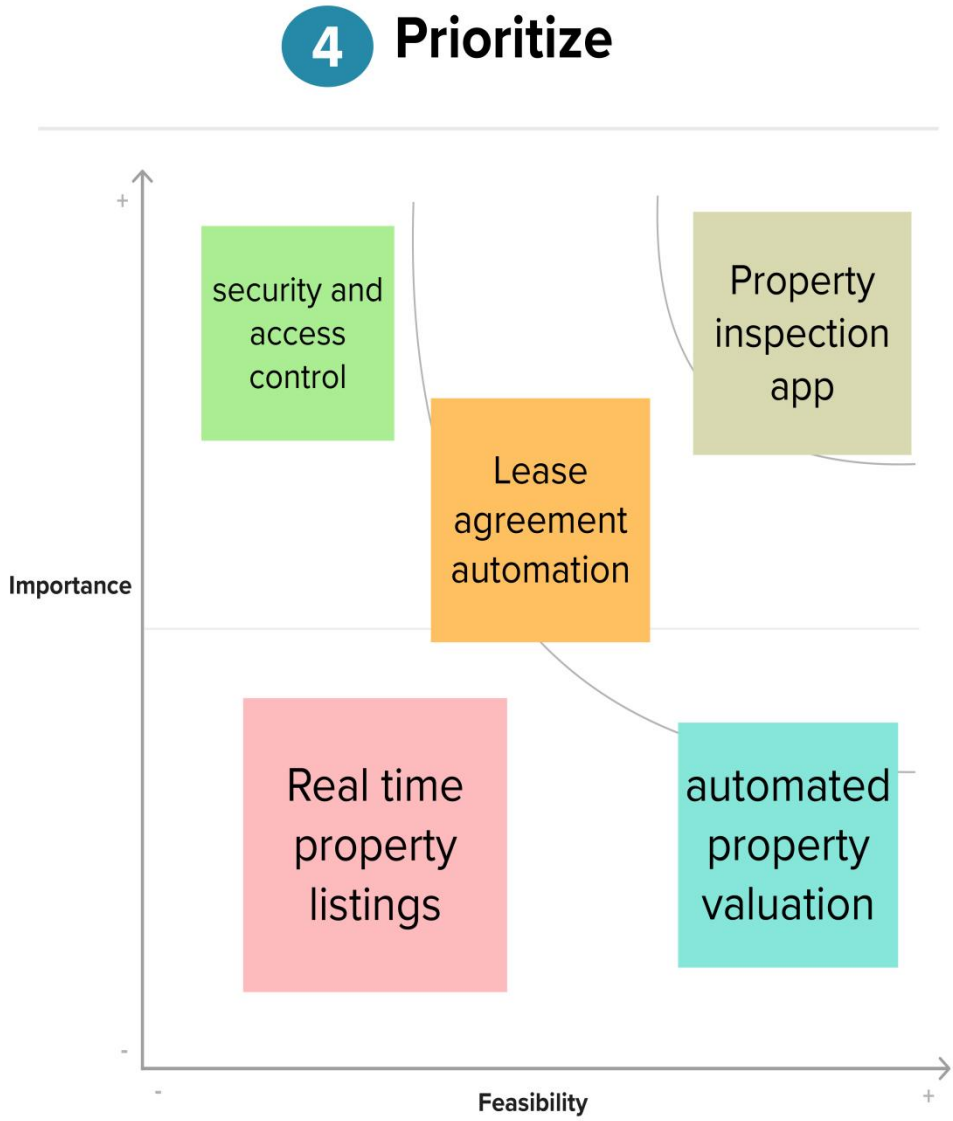
BRAINSTORM


2
Brainstorm
Maintenance request tracking
Tenant screening and management
automated property valuation
Arunadevi
Nandhini
Lease agreement automation
property management dashboard
Energy efficiency optimization
Vacancy reduction strategy
Real time property listings
security and access control
Perezhil
Mahalakshmi
FInancial forecasting and analysis
Property inspection app
Remote property monitoring

GROUP IDEAS


3
Group ideas
Maintenance request tracking-Build a mobile app for tenants to easily sumbit maintenance requests and for property managers to efficiently assign and track repairs,improving the overall maintenance process.
Vacancy reduction strategy-Develop a marketing and analytics tool that helps property owners identify factors leading to vacancies and suggest strategies for reducing them.
Property management dashboard-Design a comprehensive dashboard that property managers can use to oversee multiple properties ,including rent collection ,maintenance schedules ,and financial reports.

PRIORITIZE


4 Prioritize
+
security and access control
Property inspection app
Lease agreement automation
Importance
Real time property listings
automated property valuation
-
-
Feasibility
+

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

- **User Authentication and Authorization:**
  User registration and login with secure authentication methods.

- **Property Listing and Management:**
  Add, edit, and remove property listings with details such as location, price, size, and images.

- **Smart Contracts:**
  Implement smart contracts for property transactions, including purchase agreements, rental contracts, and lease agreements.

- **Blockchain Integration:**
  Utilize a blockchain platform (e.g., Ethereum) for storing property records and transaction history.
  Ensure data immutability and transparency by recording property transactions on the blockchain.

- **Digital Identity Verification:**
  Implement a secure system for verifying the identity of property owners and tenants.

- **Payment and Financial Transactions:**
  Enable secure payment processing for property purchases, rentals, and other transactions.
  Integration with cryptocurrency or digital payment options.

- **Document Management:**
  Store and manage property-related documents such as contracts, deeds, and inspection reports.
  Ensure the authenticity and security of documents through blockchain encryption.

- **Notifications and Alerts:**

Send notifications and alerts to users regarding property updates, contract expirations, and other relevant information.

- **Reporting and Analytics:**

Generate reports and analytics on property performance, market trends, and financial transactions.

- **Compliance and Regulatory Features:**

Ensure compliance with local real estate laws and regulations.

Allow for easy retrieval of historical data and transaction records for audits.

- **Security and Privacy:**

Implement robust security measures to protect user data and financial information.

Ensure data privacy and compliance with data protection regulations.

- **Mobile and Web Accessibility:**

Develop user-friendly web and mobile applications for easy access to the system.

These functional requirements lay the foundation for a blockchain-based real estate management system that can offer transparency, security, and efficiency in property transactions and management. Customization and additional features may be required based on specific business needs and regulatory requirements.

4.2 NON FUNCTIONAL REQUIREMENT

- **Access Control:**

Implement robust access control mechanisms to prevent unauthorized access to sensitive data.

- **Encryption:**

Encrypt data at rest and in transit to protect against data breaches.

- **Privacy Compliance:**

Ensure compliance with data privacy regulations, such as GDPR, if applicable.

- **Horizontal Scalability:**

The system should be able to handle a growing number of properties, users, and transactions without a significant decrease in performance.

- **Transaction Throughput:**

Ensure that the system can process a high volume of real estate transactions efficiently, with low latency.

- **Response Time:**

Minimize response times for user interactions, such as property searches and contract execution.

- **High Availability:**

Ensure the system is available 24/7 to accommodate users from different time zones.

- **Redundancy**:

Implement redundancy measures to minimize downtime due to hardware failures or network issues.

- **Fault Tolerance:**

The system should continue to function and maintain data integrity even in the presence of network disruptions, hardware failures, or malicious attacks.

- **User Experience (UX):**

Provide an intuitive and user-friendly interface for property owners, buyers, and tenants to interact with the system.

- **Smart Contract Efficiency:**

Optimize the execution of smart contracts to minimize gas fees and energy consumption if using a proof-of-work blockchain.

- **Resource Efficiency:**

Minimize resource consumption (e.g., energy, storage, computational power) to reduce the environmental impact if using a proof-of-work blockchain.

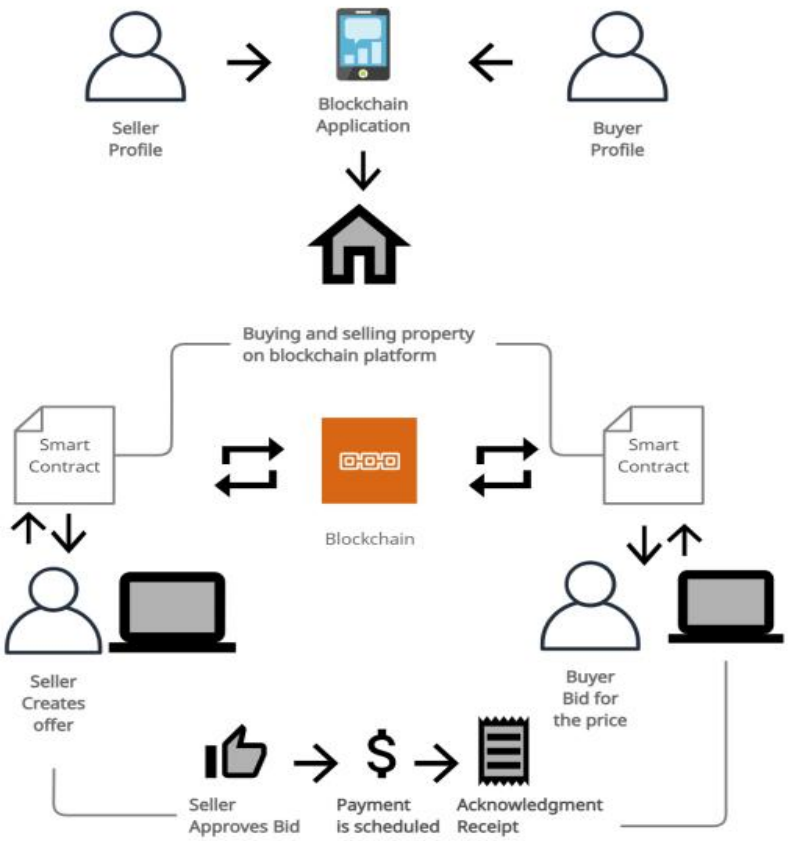
- **Comprehensive Documentation:**

Provide clear and detailed documentation for users, administrators, and developers to understand and work with the system effectively.

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS & USER STORIES

DATA FLOW DIAGRAM



USER STORY - 1 (AS A PROPERTY OWNER)

- I want to list my property on the blockchain so that I can reach potential buyers or renters securely and transparently.
- I want to verify the ownership of a property on the blockchain to prove its legitimacy to potential buyers.
- I want to receive instant notifications and updates about property showings, offers, and transaction progress.
- I want to be able to request property inspections, view inspection reports, and track maintenance history on the blockchain.
- I want access to comprehensive, easily understandable documentation and support resources to help me use the blockchain system effectively.

USER STORY - 2 (AS A BUYER)

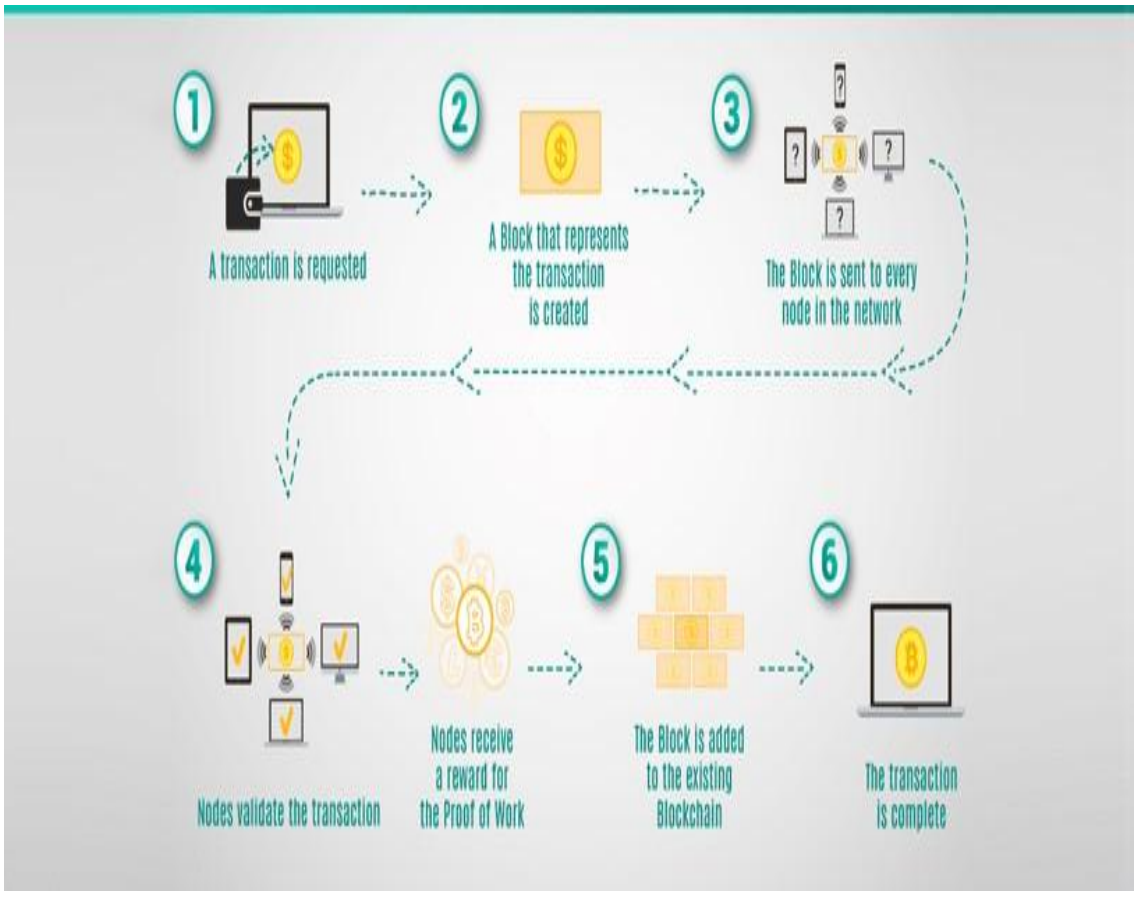
- I want to search for properties on the blockchain by location, price range, and other criteria, so I can find properties that match my needs.
- I want to see the complete transaction history of a property, including its previous owners, sale prices, and maintenance records, to make an informed purchase decision.
- I want to sign real estate contracts digitally through blockchain-based smart contracts to automate the transaction process and reduce paperwork.
- I want to be able to track the progress of my property purchase or rental agreement through real-time updates on the blockchain.
- I want a user-friendly interface with clear instructions and tooltips to help me navigate and use the blockchain-based real estate management system.
- I want the blockchain system to comply with data protection and privacy laws to ensure the security of my personal information and transaction history.

USER STORY - 3 (AS A TENANT)

- I want to search for rental properties on the blockchain and view details like monthly rent, lease terms, and property photos.

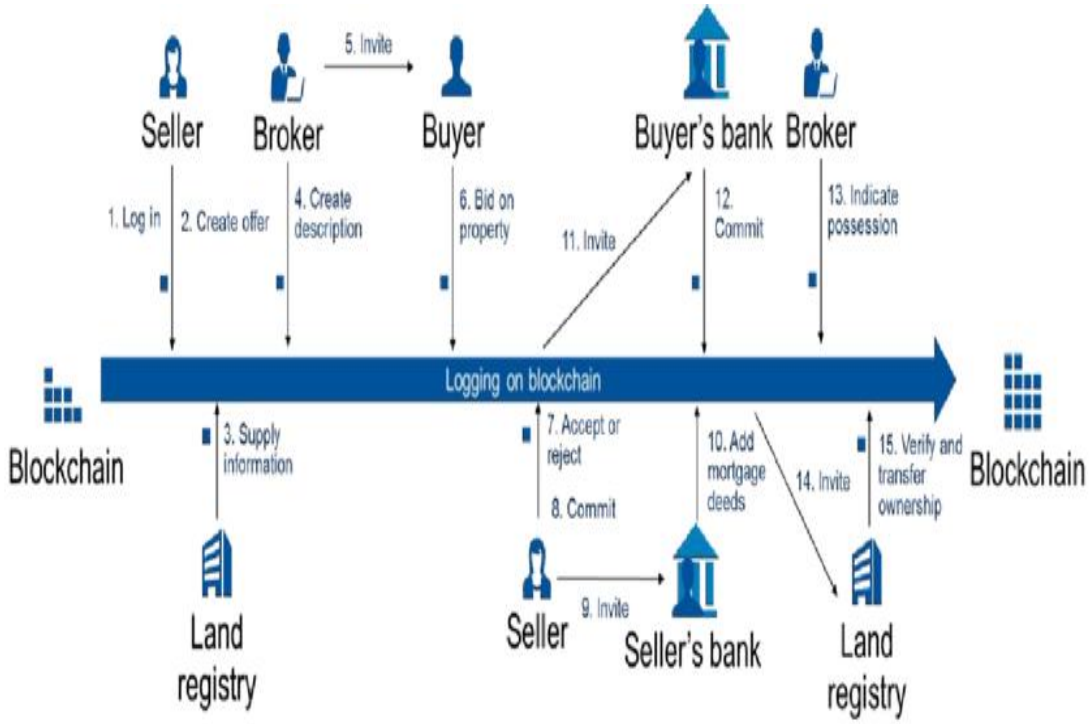
- I want to sign real estate contracts digitally through blockchain-based smart contracts to automate the transaction process and reduce paperwork.
- I want to pay my monthly rent using cryptocurrency or digital payments through the blockchain, making the process convenient and secure.

5.2 SOLUTION ARCHITECTURE



6. PROJECT PLANNING & SCHEDULING

6.1 TECHNICAL ARCHITECTRE



6.2 SPRINT PLANNING & ESTIMATION

Sprint mainly aims in selecting the user stories, tasks for user stories and the estimation time associated with it.

USER STORY - 1 (AS A PROPERTY OWNER)

- I want to list my property on the blockchain so that I can reach potential buyers or renters securely and transparently.
- I want to verify the ownership of a property on the blockchain to prove its legitimacy to potential buyers.
- I want to receive instant notifications and updates about property showings, offers, and transaction progress.
- I want to be able to request property inspections, view inspection reports, and track maintenance history on the blockchain.

- I want access to comprehensive, easily understandable documentation and support resources to help me use the blockchain system effectively.

TASK AND TASK ESTIMATION:

S.NO	TASK	TASK ESTIMATION
1.	Research and Platform Selection	2-4 weeks
2.	System Architecture Design	3-5 weeks
3.	Smart Contract Development	6-8 weeks
4.	Blockchain Integration	4-6 weeks
5.	User Registration and Property Listing	4-6 weeks
6.	Ownership Verification	4-6 weeks
7.	Notification System	6-8 weeks
8.	Testing and Quality Assurance	8-12 weeks
9.	Deployment and Launch	4-6 weeks

USER STORY - 2 (AS A BUYER)

- I want to search for properties on the blockchain by location, price range, and other criteria, so I can find properties that match my needs.
- I want to see the complete transaction history of a property, including its previous owners, sale prices, and maintenance records, to make an informed purchase decision.
- I want to sign real estate contracts digitally through blockchain-based smart contracts to automate the transaction process and reduce paperwork.
- I want to be able to track the progress of my property purchase or rental agreement through real-time updates on the blockchain.
- I want a user-friendly interface with clear instructions and tooltips to help me navigate and use the blockchain-based real estate management system.

TASK AND TASK ESTIMATION:

S. NO	TASK	TASK ESTIMATION
1.	User Interface Design	6-10 weeks
2.	Property Search Functionality	6-8 weeks
3.	Transaction History	4-6 weeks
4.	Smart Contract Integration	6-10 weeks
5.	Testing and Quality Assurance	8-12 weeks

6.	Deployment and Launch	4-6 weeks

6.3 SPRINT DELIVERY SCHEDULE

Sprint 1 - Setting the Foundation (Week 1-2):
Sprint Goal: Set up the basic infrastructure and blockchain network for the real estate system.

- ✧ Define the system architecture, technology stack, and blockchain platform (e.g., Ethereum, Hyperledger).
- ✧ Set up the development environment, including blockchain nodes and test networks.
- ✧ Establish basic smart contracts for property creation and ownership tracking.
- ✧ Develop user authentication and access control systems.
- ✧ **Delivery**: A functioning blockchain network with basic user authentication and property creation features.

Sprint 2 - Property Listing and Verification (Week 3-4):
Sprint Goal: Enable property owners to list their properties on the blockchain and verify property ownership.

- ✧ Develop property listing functionality, allowing owners to add property details, images, and pricing.
- ✧ Implement property ownership verification by integrating with external data sources, such as government land registries.
- ✧ **Delivery**: Property owners can list their properties on the blockchain, and ownership can be verified.

Sprint 3 - Smart Contracts and Transactions (Week 5-6):
Sprint Goal: Create smart contracts for property transactions and enable secure transactions on the blockchain.

- ✧ Develop and test smart contracts for property sale agreements, rental contracts, and lease agreements.

- ✧ Implement the execution of property transactions using blockchain-based smart contracts.
- ✧ **Delivery**: Users can execute property transactions securely using smart contracts.

Sprint 4 - User-Facing Features (Week 7-8):
Sprint Goal: Develop user interfaces for property listing, searching, and transaction management.

- ✧ Create web and mobile interfaces for property owners, buyers, tenants, and agents.
- ✧ Implement user-friendly features like property searching and contract management.
- ✧ **Delivery**: User interfaces are available for property-related actions.

Sprint 5 - Data Security and Compliance (Week 9-10):
Sprint Goal: Enhance data security and ensure regulatory compliance within the system.

- ✧ Implement data encryption for sensitive information at rest and in transit.
- ✧ Ensure compliance with data privacy regulations and real estate laws.
- ✧ **Delivery**: Data is more secure, and the system is in compliance with relevant regulations.

Sprint 6 - Integration and Testing (Week 11-12):
Sprint Goal: Integrate external data sources, perform extensive testing, and prepare for deployment.

- ✧ Integrate with external property databases, government systems, and payment gateways.
- ✧ Conduct thorough testing, including unit testing, integration testing, and user acceptance testing.
- ✧ Prepare for production deployment.
- ✧ **Delivery**: A thoroughly tested and ready-to-deploy system.

7. CODING & SOLUTIONING

7.1 FEATURE 1 - SMART CONTRACT

Smart contracts play a crucial role in a real estate management system using blockchain by automating and securing various aspects of real estate transactions. Here are some essential smart contract features that can be incorporated into such a system

1. Property Listing and Ownership Verification:

Smart contracts to facilitate property listings and verify property ownership. When a property is listed, the smart contract records the owner's information, making it tamper-proof and transparent.

2. Digital Contract Signing:

Smart contracts for digital contract signing between buyers and sellers. These contracts can include terms, conditions, and legal obligations, ensuring automated and secure agreements.

3. Payment Handling:

Smart contracts for handling and disbursing payments securely. These contracts can automatically release funds to the seller upon meeting predefined conditions, such as successful property transfer.

4. Ownership Transfer:

Smart contracts to manage the transfer of property ownership, updating the ownership records on the blockchain when all conditions of the contract are met.

5. Automated Notifications:

Smart contracts to send automated notifications to involved parties (buyers, sellers, agents) at key milestones of the transaction, such as contract signing, inspections, and successful transfer.

6. Record Keeping:

Smart contracts that maintain a secure and unchangeable record of all real estate transactions, including details of the parties involved, sale prices, dates, and other transaction data.

7. Tokenization of Real Estate:

Smart contracts for dividing real estate properties into tradable tokens, allowing for fractional ownership and easier investment in real estate.

8. Estate Planning and Inheritance:

Smart contracts to facilitate estate planning, allowing property owners to specify inheritance arrangements and automating the transfer of property upon specific events, such as the owner's passing.

9. Time-locked Contracts:

Implement smart contracts with time-locked features for scheduled disbursements or property transfers, ensuring that certain actions occur at specified future dates.

10. Privacy and Data Protection:

Implement smart contracts that protect sensitive personal data and transaction details while ensuring compliance with data protection laws.

7.2 FEATURE 2 - METAMASK

MetaMask is a popular Ethereum wallet and gateway to the decentralized web. When integrating MetaMask into a real estate management system using blockchain, you can leverage several features and capabilities to enhance the user experience and facilitate secure interactions with the Ethereum blockchain. Here are some key features of MetaMask and how they can be applied in the context of a real estate management system:

1. User Wallet Management:

Users can create and manage their Ethereum wallets through MetaMask, allowing them to securely store their digital assets and interact with the real estate management system.

2. Secure Transactions:

MetaMask provides a secure environment for users to initiate and authorize transactions, such as property purchases, contract signings, and payments, ensuring the security of their assets and sensitive data.

3. Web3 Integration:

MetaMask supports the Web3 JavaScript API, which allows your real estate management system to connect to users' wallets and request signatures for transactions and contract interactions.

4. Transaction History:

Users can view their transaction history within MetaMask, enabling them to track their real estate-related activities on the blockchain.

5. Token Management:

MetaMask allows users to manage Ethereum-based tokens, including Ether (ETH) and any property-related tokens or NFTs (e.g., ERC-20 or ERC-721 tokens representing real estate assets).

6. Connectivity with DApps:

MetaMask facilitates interactions with decentralized applications (DApps). Your real estate management system can function as a DApp, enabling users to access and utilize your platform seamlessly.

7. Private Keys and Recovery Phrases:

MetaMask helps users manage private keys and recovery phrases, allowing them to recover access to their wallets if necessary. It's important to educate users about the importance of safeguarding these keys.

8. Security and Privacy:

MetaMask prioritizes user security and privacy. Integrating MetaMask into your system can provide users with confidence in the security of their transactions and data.

9. Gas Fee Management:

Inform users about gas fees associated with Ethereum transactions and how to adjust gas settings within MetaMask to ensure efficient and cost-effective interactions with your system.

10. User-Friendly UI/UX:

Ensure a seamless and user-friendly interface between your real estate management system and MetaMask, simplifying user interactions and reducing friction.

11. Transaction Confirmation:

MetaMask provides a clear and transparent way for users to confirm and authorize transactions, which is essential for real estate-related operations like property purchases and contract signings.

8. PERFORMANCE TESTING

8.1 PERFORMANCE METRICS

1. Transaction Throughput:

Measure the number of property transactions the system can handle per unit of time (e.g., transactions per second or per minute). This metric assesses the system's capacity for processing real estate transactions efficiently.

2. Transaction Confirmation Time:

Track the time it takes for transactions to be confirmed and added to the blockchain. Users value quick transaction confirmations, especially in real estate transactions, where time can be critical.

3. Scalability:

Assess the system's ability to scale as the number of users and property listings grows. Evaluate how well it handles increased loads without a significant drop in performance.

4. System Response Time:

Measure the time it takes for the system to respond to user requests, such as property searches, contract signings, and maintenance requests. Quick response times are crucial for user satisfaction.

5. Latency:

Monitor network latency to ensure that data transfer between users, the blockchain, and external systems is not causing delays. Low latency is essential for real-time updates and interactions.

6. Gas Fees:

Analyze the cost of gas fees associated with each transaction. High fees may discourage users from using the system, so optimizing gas fees is important.

7. Error Rate:

Measure the rate of errors, failed transactions, or system issues. Reducing the error rate is crucial for ensuring reliability and user trust.

8. Compliance and Legal Metrics:

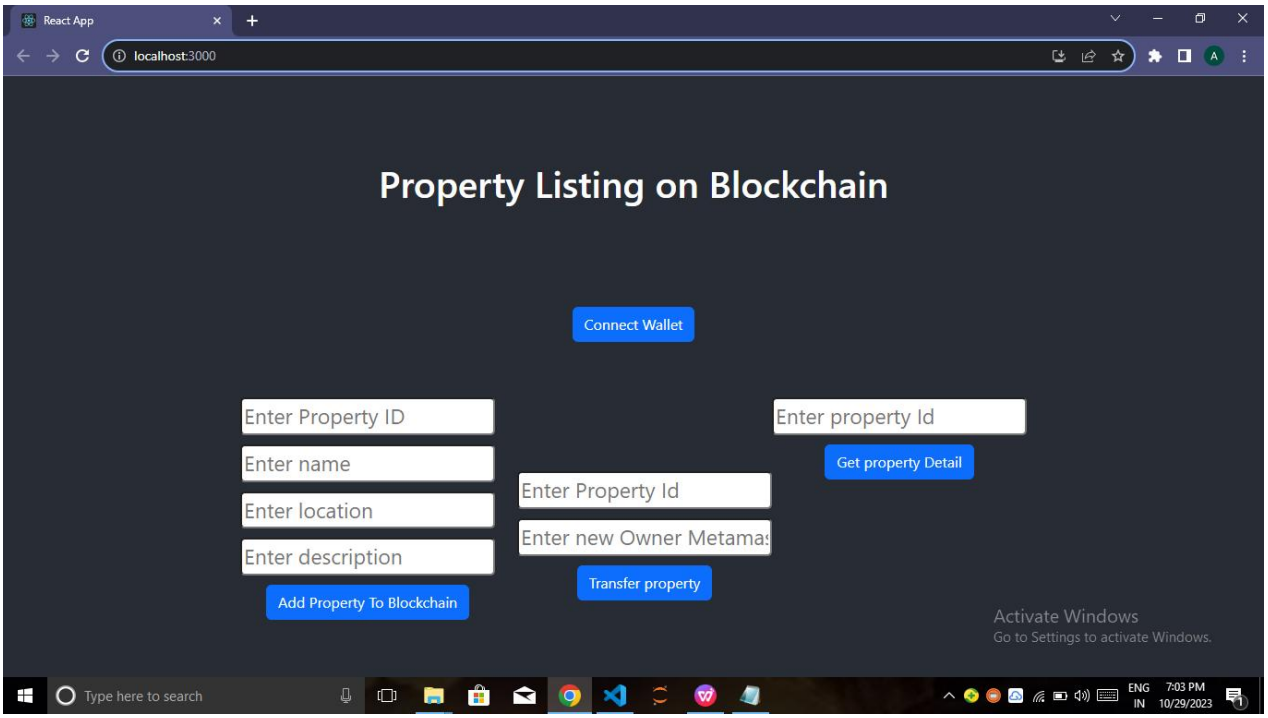
Ensure that the system complies with legal and regulatory requirements, and monitor metrics related to adherence to data protection and privacy laws.

9. Support and Ticket Metrics:

Analyze the number and types of user support tickets and inquiries. This can indicate areas of the system that may need improvement or clarification.

9.RESULTS

9.1 OUTPUT SCREENSHOTS



10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Transparency and Trust
- Reduced Fraud
- Smart contracts
- Decentralization
- Faster Transactions
- Lower Costs
- Accessibility
- Fractional Ownership
- Data Security
- Improved Due Diligence
- Property Tokenization
- Compliance and Regulation

DISADVANTAGES

- Limited Adoption
- Regulatory and Legal Hurdles
- Privacy Concerns
- Lack of Interoperability
- Technical Complexity
- Risk of Data Loss
- Initial Costs
- Resistance to Change
- Energy Consumption
- Security Concerns

11. CONCLUSION

In conclusion, a real estate management system using blockchain technology has the potential to revolutionize the real estate industry, offering several significant advantages, but it also comes with notable challenges.

The successful implementation of a blockchain-based real estate management system requires a careful evaluation of the specific needs, regulatory environment, and technological readiness of the real estate market in question.

As the technology matures and the industry evolves, some of the current disadvantages may become more manageable, making blockchain an increasingly viable solution for real estate management.

12. FUTURE SCOPE

Increased Adoption:

As blockchain technology matures and becomes more widely accepted, it is likely to see increased adoption within the real estate industry. More companies, governments, and institutions may start using blockchain for property management, transactions, and record-keeping.

Global Investment:

Blockchain can facilitate cross-border real estate investments. Investors from different parts of the world can easily access and invest in properties in other countries, increasing diversification and expanding opportunities.

Smart Cities:

Smart city initiatives, which integrate technology to enhance urban living, can benefit from blockchain in real estate management. Property data, energy management, and utility usage can be efficiently and securely handled on a blockchain.

Integration with IoT:

The Internet of Things (IoT) can be combined with blockchain for real estate management to improve security, maintenance, and energy

efficiency. For example, IoT sensors can monitor property conditions and automatically trigger maintenance or notify property managers of issues.

Blockchain-Based Property Records:

The use of blockchain for property records and title management can enhance the integrity of land titles, reducing disputes and legal complications.

Sustainability and Green Real Estate:

Blockchain can be used to verify and record the sustainability and energy efficiency of real estate properties, contributing to the green real estate market.

Property Crowdfunding:

Blockchain enables fractional ownership, allowing individuals to invest in real estate through crowdfunding platforms. This opens up opportunities for a broader range of investors to participate in the real estate market.

13. APPENDIX

SOURCE CODE

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

contract PropertyDetail{
   address public owner;

   struct Property {
      string propertyId;
      string name;
      string location;
      string discription;
      address currentOwner;
   }

   mapping(string => Property) public properties;
   mapping(address => mapping(string => bool)) public hasAccess;
```

```
event PropertyAdded(
    string indexed propertyId,
    string name,
    string location,
    address indexed owner
);
event PropertyTransferred(
    string indexed propertyId,
    address indexed from,
    address indexed to
);

constructor() {
    owner = msg.sender;
}

modifier onlyOwner() {
    require(msg.sender == owner, "Only contract owner can call this");
    _;
}

modifier hasPropertyAccess(string memory propertyId) {
    require(
        hasAccess[msg.sender][propertyId],
        "You don't have access to this property"
    );
    _;
}

function addProperty(
    string memory propertyId,
    string memory name,
    string memory location,
    string memory _description
) external onlyOwner {
    require(
        bytes(properties[propertyId].propertyId).length == 0,
        "Property already exists"
```

```
        );

        properties[propertyId] = Property({
            propertyId: propertyId,
            name: name,
            location: location,
            discription : _description,
            currentOwner: owner
        });

        hasAccess[owner][propertyId] = true;

        emit PropertyAdded(propertyId, name, location, owner);
    }

    function transferProperty(
        string memory propertyId,
        address newOwner
    ) external hasPropertyAccess(propertyId) {
        require(newOwner != address(0), "Invalid new owner");

        address currentOwner = properties[propertyId].currentOwner;
        properties[propertyId].currentOwner = newOwner;

        hasAccess[currentOwner][propertyId] = false;
        hasAccess[newOwner][propertyId] = true;

        emit PropertyTransferred(propertyId, currentOwner, newOwner);
    }

    function getPropertyDetails(
        string memory propertyId
    ) external view returns (string memory, string memory, address) {
        Property memory prop = properties[propertyId];
        return (prop.name, prop.location, prop.currentOwner);
    }
}
```

GITHUB LINK:

https://github.com/Nandhini-V-R/NM-Blockchain-RealEstate.git

DEMO LINK:

https://drive.google.com/file/d/1GuFqA4HOZ19YawLEyGeDeVkHJ69q3o5s/view?usp=drivesdk