FARMER INSURANCE CHAIN

Project Report

Submitted by:

TEAM ID :
NM2023TMID00512

G.K.M COLLEGE OF ENGINEERING AND TECHNOLOGY CHENNAI-600063

TEAM MEMBERS:

SHALINI. V - 410820121034

KIRUBHARATHI. R - 410820121024

ELAKKIYA. S - 410820121010

NIVETHA. S - 410820121029

TABLE OF CONTENT

CHAPTER – 1

1.INTRODUCTION

1.1 Project Overview

Farmer insurance chain

The Farmers' Insurance is an American insurance company that offers a wide range of insurance products, including auto, home, life, and business insurance. It's known for its distinctive red umbrella logo. The company provides coverage to millions of customers across the United States and offers various policy options to meet different insurance needs.Farmers Insurance is a prominent American insurance company with a history dating back to 1928. It provides a wide array of insurance products and services, including auto, home, renters, life, and business insurance. Farmers is known for its extensive network of agents and a range of coverage options to meet the diverse needs of its customers. The company emphasizes personalized service and has a strong presence in the insurance market, serving millions of policyholders across the United States.

1.2 Purpose

The purpose of a farmer insurance chain is to provide insurance services and products tailored specifically for farmers and agricultural businesses. These insurance offerings are designed to protect farmers from various risks and uncertainties they face in their operations, including crop losses, equipment damage, liability issues, and more. The chain's purpose is to support the agricultural industry by helping farmers manage and mitigate these risks, ensuring the sustainability and stability of their businesses.

CHAPTER - 2
2.LITERAURE SURVEY

2.1 Existing Problem

The existing problems in the farmer insurance chain can vary depending on the region and the specific circumstances, but some common issues include:

1.Limited access to insurance: Many farmers, especially in developing countries, lack access to insurance products tailored to their needs. This leaves them vulnerable to financial losses due to crop failures, natural disasters, or other risks.

2.High premium costs: In some cases, the cost of insurance premiums can be prohibitive for small-scale farmers, making it challenging for them to afford coverage.

3.Lack of data and technology: Insufficient data and outdated technology can hinder the efficient management of insurance claims and risk assessment for farmers.

4.Delayed or inadequate payouts: Some farmers experience delays or receive inadequate payouts when they file insurance claims, which can disrupt their livelihoods and recovery efforts.

5.Weather-related uncertainties: Weather-dependent insurance products may be unreliable due to increasingly unpredictable and extreme weather patterns caused by climate change.

2.2 Reference

https://www.mdpi.com/2071-1050/13/16/8921

https://www.indexinsuranceforum.org/blog/blockchain-application-agriculture-insurance

https://acreafrica.com/reimagining-agriculture-insurance-using-blockchain-technology/

https://www.hindawi.com/journals/acisc/2022/8011525/

https://www.researchgate.net/publication/344708973_Blockchain_Based_Crop_Insurance_A_Decentralized_Insurance_System_for_Modernization_of_Indian_Farmers

2.3 Problem Statement Definition:

The “Farmer Insurance Chain Problem” statement is not a well-known or widely recognized term or issue in the field of insurance or agriculture as of my last knowledge update in January 2022. To provide a meaningful definition, I would need more context or information about this specific problem. If it’s a newly emerging issue or a specialized topic, I recommend providing additional details or clarifications so I can better assist you.

CHAPTER-3

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas :

Creating an empathy map for a project involving blockchain technology in the context of farmer insurance requires understanding the various stakeholders involved and their perspectives. Here’s a simplified empathy map:

1. **Says**:

- The Farmers: They may express concerns about the complexity of the insurance process, the delay in claim settlements, and the need for affordable coverage.

- Insurance Companies: They might voice issues about fraud prevention and operational efficiency.

2. **Thinks**:

- Farmers: They might be thinking about the unpredictability of weather and crop yields and how insurance can help them manage risks.

- Insurance Companies: They may be considering how blockchain technology can streamline claim processing and reduce administrative costs.

3. **Does**:

- Farmers: They may seek information on insurance options, apply for policies, and file claims.

- Insurance Companies: They process applications, assess claims, and maintain records.

4. **Feels**:

- Farmers: They may feel anxious about crop losses and relieved when they have insurance coverage.

- Insurance Companies: They might feel the need for trust and transparency in claim settlements.

5. **Pain Points**:

- Farmers: Delayed claim settlements, lack of understanding of insurance terms and conditions, high premiums.

- Insurance Companies: Fraudulent claims, administrative overhead, lack of transparency.

6. **Gains**:

- Farmers: Peace of mind, financial stability during crop failures, trust in the insurance system.

- Insurance Companies: Reduced fraud, lower operational costs, increased customer trust.

This empathy map helps identify the concerns, motivations, and pain points of both farmers and insurance companies involved in the blockchain-based farmer insurance chain project. It can guide the design and implementation of the project to better meet the needs of these stakeholders.


Says
What have we heard them say?
What can we imagine them saying?
Insurance Experts
Regulatory Authorities
Agricultural Community
*Blockchain Developers
Thinks
What are their wants, needs, hopes, and dreams?
What other thoughts might influence their behavior?
Policy Makers
Industry Insights
Stakeholder Perspectives
Risk Assessment
V.SHALINI
R.KIRUBHARATHI
S.ELAKKIYA
S.NIVETHA
Field Testing
Data Verification
System Integration
Blockchain Implementation
Stakeholder Sentiment
User Experience
Project Confidence
Community Reaction
Does
What behavior have we observed?
What can we imagine them doing?
Feels
What are their fears, frustrations, and anxieties?
What other feelings might influence their behavior?
See an example

3.2 Ideation & Brainstorming:

Creating a blockchain-based Farmer Insurance Chain project involves utilizing blockchain technology to address challenges in the agricultural insurance industry. Here's a brainstorming outline for the project:

1. **Smart Contracts for Policies:**

- Implement smart contracts on the blockchain to automate the creation, management, and execution of insurance policies for farmers. These contracts can self-execute when predefined conditions are met.

2. **Decentralized Identity and Verification:**

- Develop a system for farmers to create and maintain their digital identities on the blockchain. This can include their land ownership, crops, and historical insurance data.

3. **Crop Monitoring and Data Feeds:**

- Integrate IoT devices and data feeds for real-time monitoring of crops, weather conditions, and other relevant data. This information can be recorded on the blockchain to assist in claim verification.

4. **Risk Assessment and Premium Calculation:**

- Use historical data and AI algorithms to assess the risk associated with different crops and regions. Calculate premiums dynamically based on this data and store it on the blockchain.

5. **Claim Processing and Payouts:**

- Enable quick and transparent claim processing through smart contracts. Farmers can submit claims, and if conditions are met, payouts are automatically triggered.

6. **Blockchain Consortium of Stakeholders:**

- Form a consortium of key stakeholders, including insurance companies, farmers, regulators, and data providers, to participate in the blockchain network.

7. **Immutable Data and Transparency:**

- Leverage the immutability of blockchain to maintain a transparent history of policies, claims, and transactions, reducing fraud and dispute resolution time.

8. **Data Privacy and Security:**

- Implement privacy-focused blockchain solutions to protect sensitive farmer data while still allowing for transparency and auditability.

9. **Mobile and User-Friendly Interface:**

- Develop user-friendly mobile applications for farmers to access and manage their insurance policies and claims conveniently.

10. **Tokenized Incentives and Payments:**

- Introduce a token-based reward system to incentivize farmers for adopting the system and sharing data. These tokens can be used for premium discounts or exchanged for other benefits.

11. **Regulatory Compliance:**

- Ensure that the project complies with relevant agricultural and insurance regulations in the target regions.

12. **Scalability and Interoperability:**

- Plan for scalability to handle a growing number of participants and interoperability with other blockchain networks and legacy systems.

13. **Educational Initiatives:**

- Launch educational programs to teach farmers about the benefits of blockchain-based insurance and how to use the platform effectively.

14. **Community Building:**

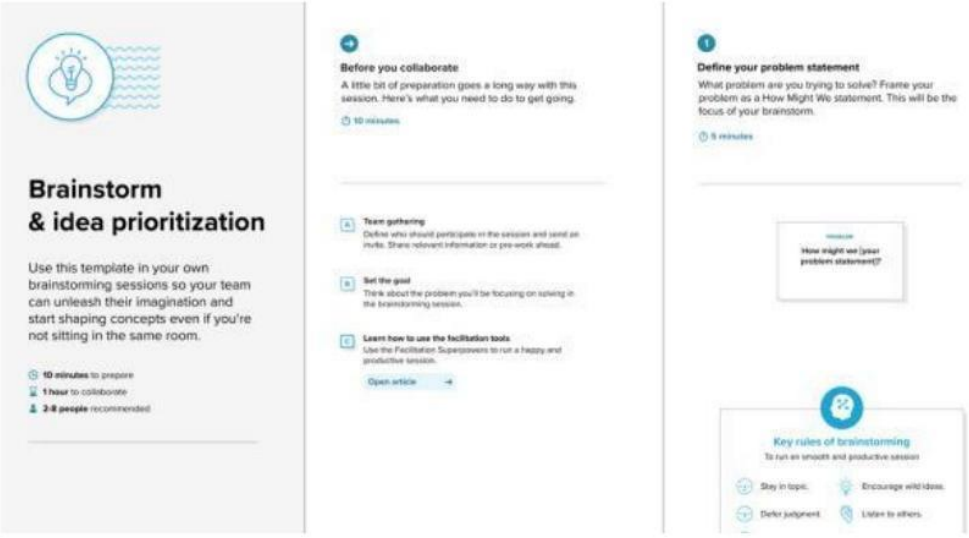
- Foster a community around the project to share experiences, address concerns, and encourage participation.

15. **Measuring Impact:**

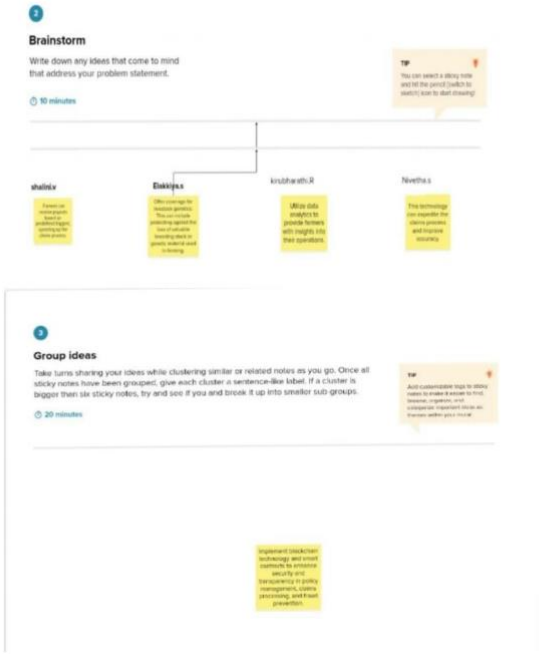
- Establish KPIs to measure the impact of the project on the financial security and well-being of farmers, as well as the efficiency of insurance operations.

Remember to conduct thorough research, collaborate with experts, and consider the specific needs and challenges of the target farming communities when implementing this project.

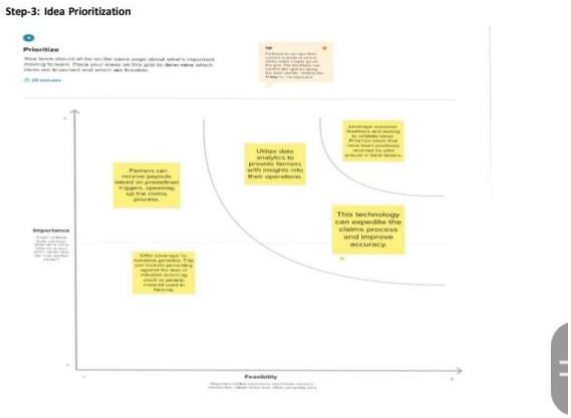
Step-1: Team Gathering, Collaboration and Select the



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



CHAPTER-4

4.REQUIREMENT ANALYSIS

4.1 Functional Requirements

The functional Functional requirements for a Farmer Insurance Chain project encompass various aspects, from user registration and policy management to smart contracts, data integration, and compliance with regulations. Farmers should have a user-friendly interface for creating and managing policies, while smart contracts automate premium payments and claims processing. Crop monitoring and risk assessment through AI algorithms ensure accurate premium calculations. Privacy and data security are crucial, with notifications and tokenized incentives to engage farmers. Mobile apps, regulatory compliance, and interoperability with other systems enhance user experience and ensure regulatory alignment.

Additionally, transparency, scalability, community support, educational resources, and reporting tools are vital components. Auditors and regulators can verify transactions, and the system should be capable of handling an expanding user base. The project's impact can be measured through defined KPIs and analytics, ensuring that the blockchain-based Farmer Insurance Chain aligns with the specific needs of the farming communities it serves.

4.2 Non-Functional Requirements

Non-functional requirements for a Farmer Insurance Chain project using blockchain are essential criteria that govern how the system performs and interacts with its users and the environment. Security is paramount, ensuring the protection of sensitive farmer data and financial transactions, guarding against unauthorized access and fraudulent activities. Performance expectations demand a responsive platform capable of efficiently handling a high volume of transactions with minimal downtime, emphasizing low latency to ensure a seamless user experience. Scalability is crucial to accommodate the project's growth, ensuring that as the user base and transaction volume expand, the system can adapt without sacrificing performance.

Reliability standards dictate that the system should be highly dependable, incorporating redundancy and failover mechanisms to maintain continuous service availability. Compliance requirements focus on adhering to data protection regulations, insurance industry standards, and agricultural regulatory frameworks in the regions served, ensuring that the project operates within legal and ethical boundaries. Interoperability is necessary for integrating with existing agricultural and insurance systems, facilitating data exchange and collaboration. Usability mandates an intuitive and user-friendly interface that minimizes the need for extensive user training, enhancing accessibility for farmers and other stakeholders.

Additionally, auditability requirements emphasize maintaining an auditable trail of all transactions and activities, supporting regulatory and auditing purposes. Data backup and recovery mechanisms should be robust to safeguard against data loss and maintain business continuity. Response time expectations define specific requirements for system functions to ensure prompt user interactions, while data integrity standards mandate that data on the blockchain remains accurate and tamper-proof over time. Resource efficiency considerations focus on optimizing the utilization of computing power, storage, and network bandwidth. Support and maintenance are essential, with ongoing support and updates to address issues and assist users effectively.

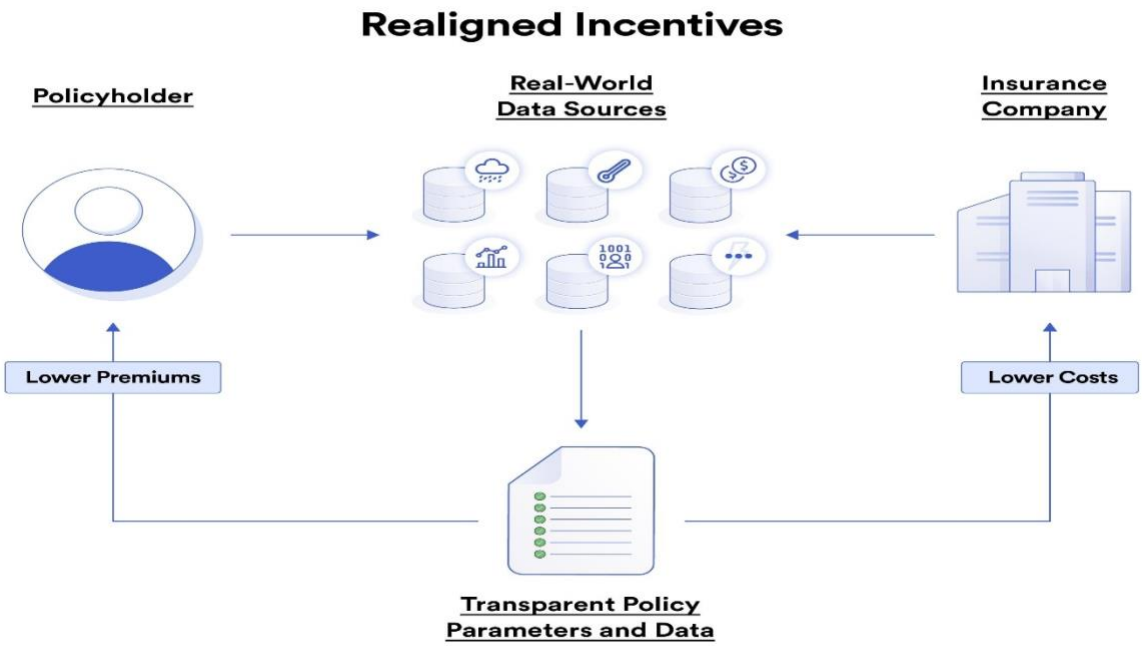
Accessibility standards ensure that the platform is accessible to all users, including those with disabilities, complying with accessibility regulations. Disaster recovery planning is critical to ensure business continuity in unforeseen events, and load testing verifies the system's ability to handle peak usage without performance degradation. Cost management is crucial, necessitating cost-

effective solutions for infrastructure, maintenance, and user support. Localization is needed if operating in multiple regions, including language and currency preferences. Data encryption requirements call for protecting sensitive data at rest and during transmission to thwart unauthorized access. Lastly, user training and documentation should be provided to ensure that users, including farmers, understand how to effectively use the system, promoting adoption and usability.

CHAPTER-5

5 .PROJECT DESIGN

5.1 Data Flow Diagram &User Stories



5.2 User Stories

Certainly, here are the user stories for a blockchain-based farmer insurance project summarized in 5 key points:

1. **Farmers' Access**: Farmers need a simple way to purchase crop insurance on a blockchain platform to protect their crops in case of failure.

2. **Insurance Agents' Efficiency**: Insurance agents require a secure and transparent blockchain system for issuing policies to farmers, reducing fraud and administrative workload.

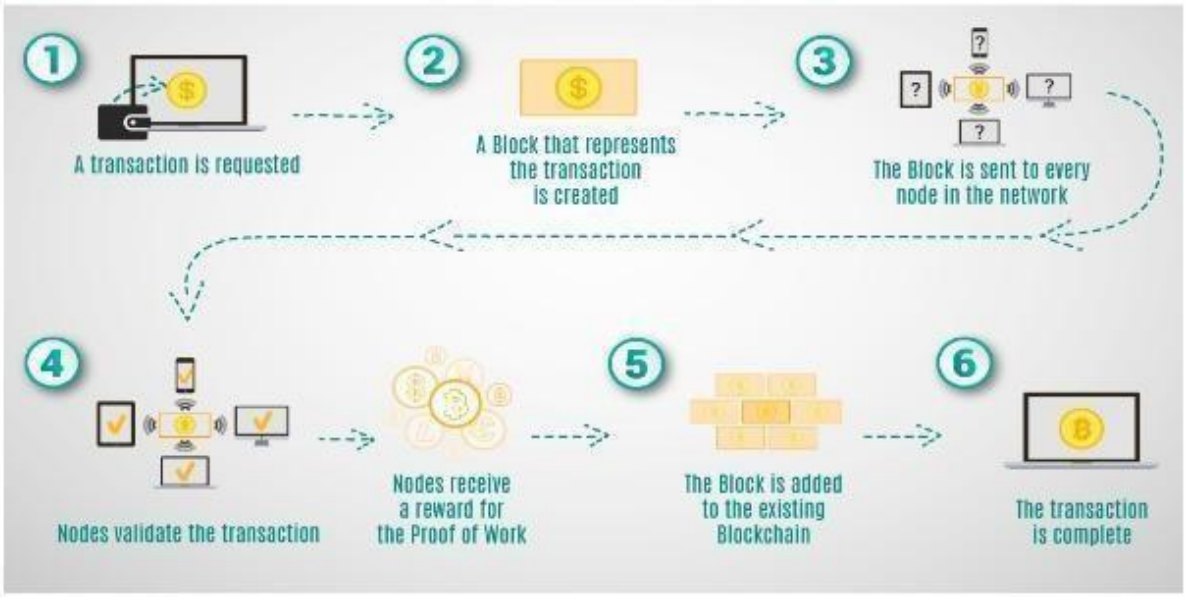
3. **Government Regulators' Data Access**: Government regulators seek real-time access to insured crop data on the blockchain to enforce policies and make informed decisions regarding agricultural subsidies.

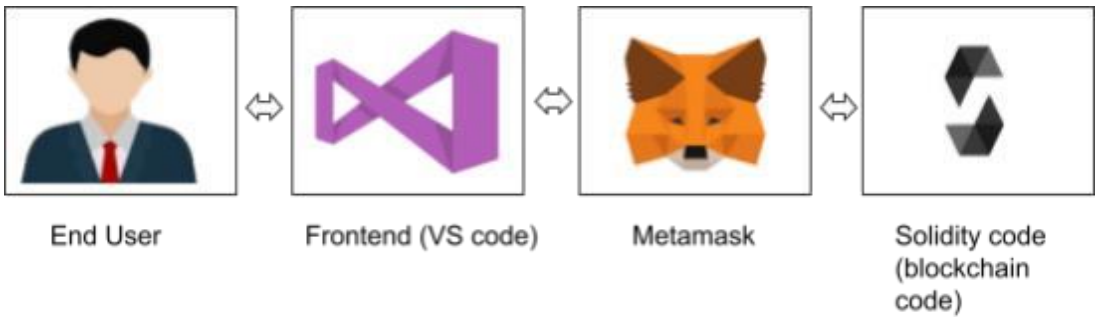
4. **Financial Institution Integration**: Financial institutions aim to provide loans to farmers based on blockchain-verified insurance coverage, minimizing risks and streamlining loan approval processes.

5. **Claims Processing Optimization**: Claims adjusters need an efficient process for handling claims through blockchain smart contracts, reducing paperwork and processing time, ensuring faster and transparent settlements

CHAPTER-6
6.PROJECT PLANNING AND SCHEDULING

6.1 Technical Architecture





CHAPTER-7 7.CODING AND SOLUTIONING

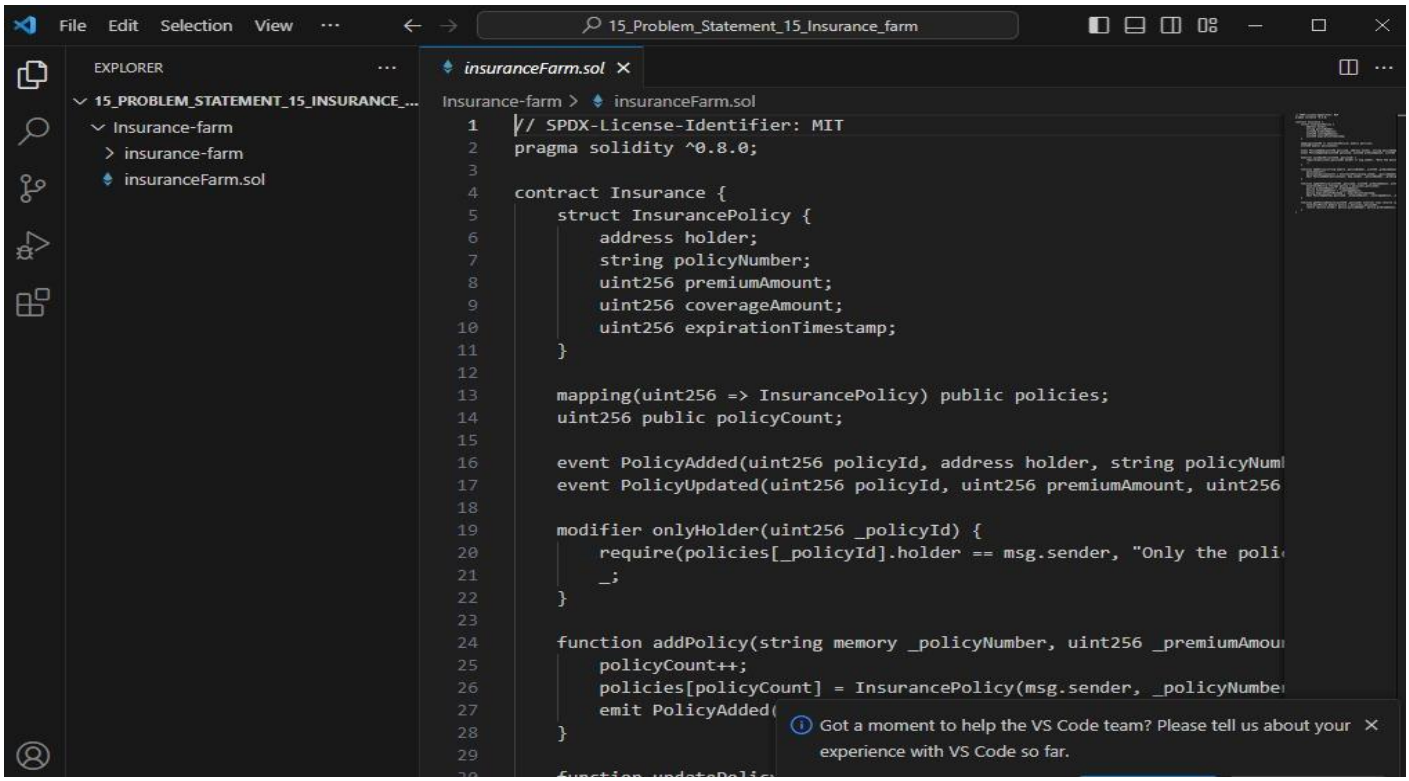
```
// SPDX-License-Identifier: MIT
Pragma solidity ^0.8.0;

Contract FarmerInsurance {
   Struct Policy {
      Address farmer;
      Uint256 premium;
      Uint256 coverage;
      Bool is Active;
   }

   Policy[] public policies;
```

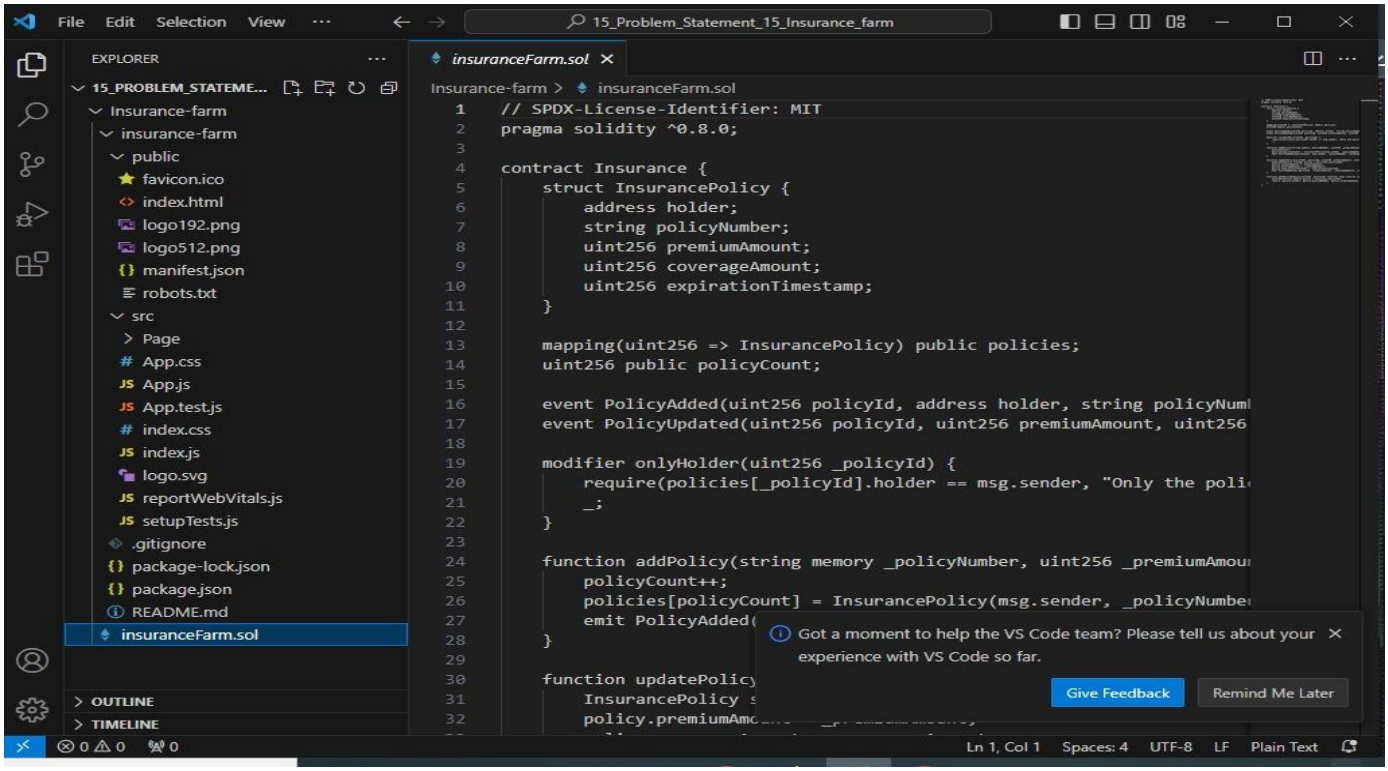
```
Address public owner;

Modifier onlyOwner() {
    Require(msg.sender == owner, “Only the owner can perform this action”);
    _;
}

Event PolicyCreated(address indexed farmer, uint256 premium, uint256 coverage);
Event ClaimSubmitted(address indexed farmer, string details);

Constructor() {
    Owner = msg.sender;
}

Function createPolicy(uint256 premium, uint256 coverage) external payable {
    Require(msg.value == premium, “Premium payment must match premium amount.”);

    Policy memory newPolicy = Policy({
        Farmer: msg.sender,
        Premium: premium,
        Coverage: coverage,
        isActive: true
    });

    Policies. Push(newPolicy);

    Emit PolicyCreated(msg.sender, premium, coverage);
}

Function submitClaim(string memory details) external {
    For (uint256 I = 0; I < policies.length; i++) {
        If (policies[i].farmer == msg.sender && policies[i].isActive) {
            // Process the claim, perform necessary checks, and disburse funds if approved.
            // You would implement claim processing logic here.

            Emit ClaimSubmitted(msg.sender, details);
            Break;
        }
    }
}
```

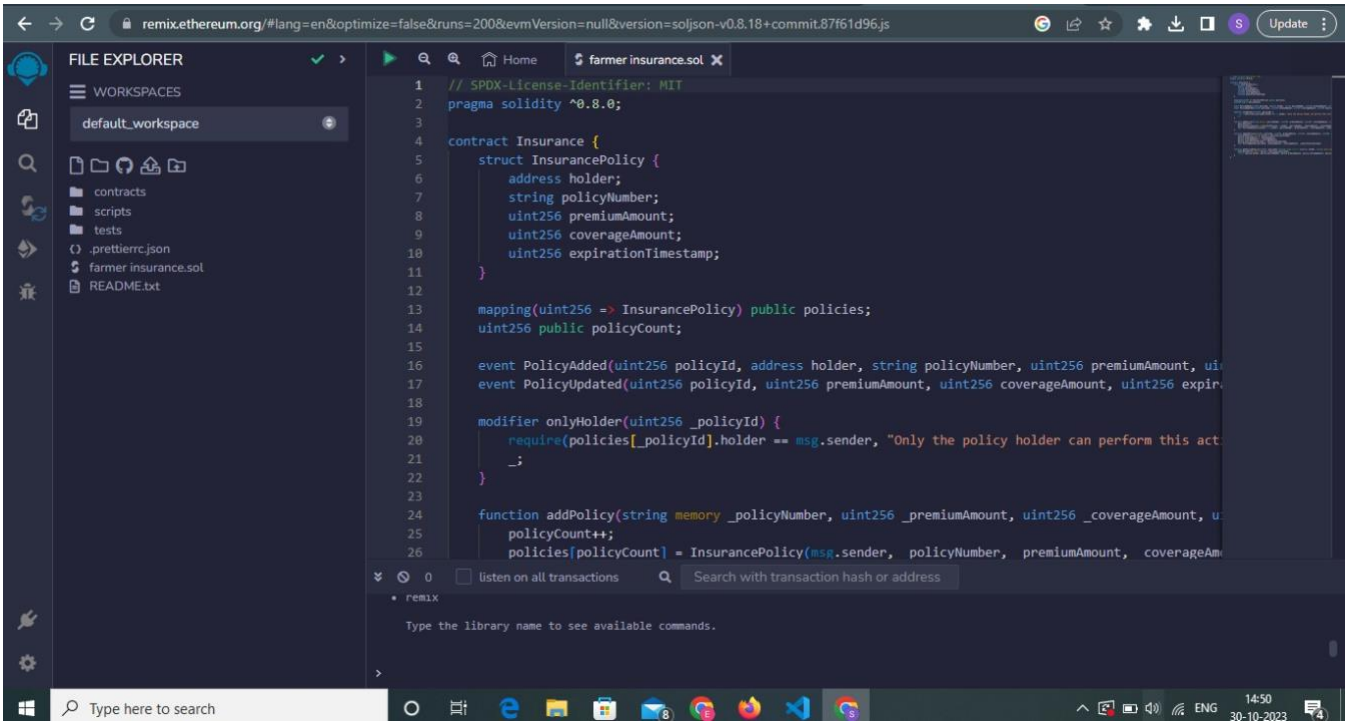
```
  Function getPolicyCount() external view returns (uint256) {
     Return policies.length;
  }
}
```

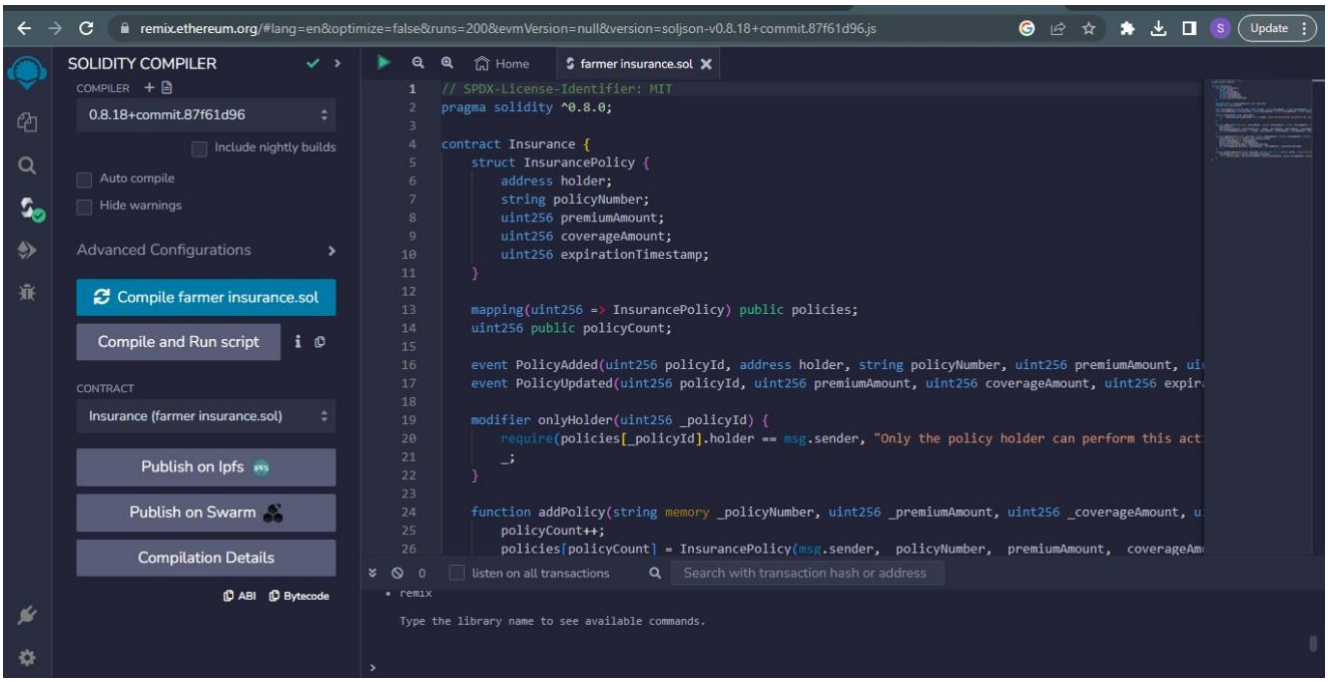
7.1 Feature 1

Visual Studio Coding



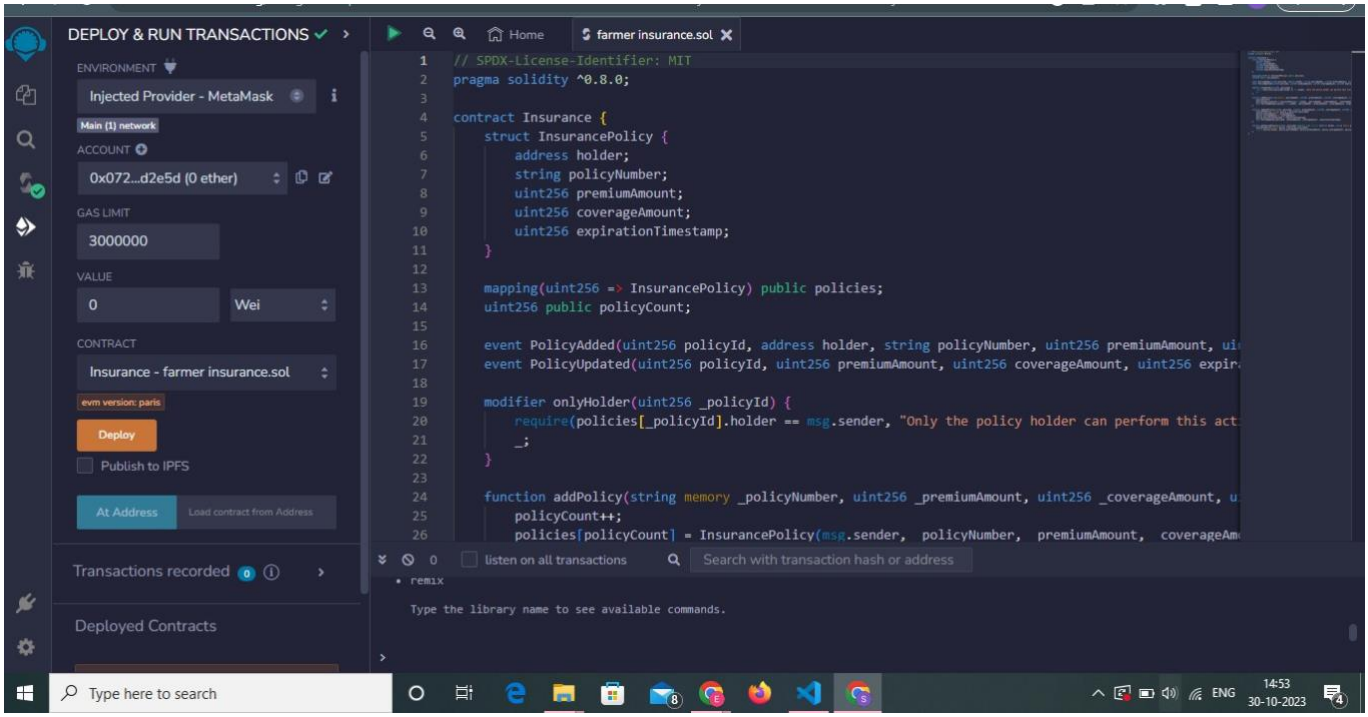

File Edit Selection View
15_Problem_Statement_15_Insurance_farm
EXPLORER
15_PROBLEM_STATEME...
Insurance-farm
insurance-farm
public
favicon.ico
index.html
logo192.png
logo512.png
manifest.json
robots.txt
src
Page
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
insuranceFarm.sol
OUTLINE
TIMELINE
insuranceFarm.sol
Insurance-farm > insuranceFarm.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Insurance {
struct InsurancePolicy {
address holder;
string policyNumber;
uint256 premiumAmount;
uint256 coverageAmount;
uint256 expirationTimestamp;
}
mapping(uint256 => InsurancePolicy) public policies;
uint256 public policyCount;
event PolicyAdded(uint256 policyId, address holder, string policyNum
event PolicyUpdated(uint256 policyId, uint256 premiumAmount, uint256
modifier onlyHolder(uint256 _policyId) {
require(policies[_policyId].holder == msg.sender, "Only the poli
_;
}
function addPolicy(string memory _policyNumber, uint256 _premiumAmou
policyCount++;
policies[policyCount] = InsurancePolicy(msg.sender, _policyNumbe
emit PolicyAdded(
}
function updatePolicy
InsurancePolicy s
policy.premiumAmo
Got a moment to help the VS Code team? Please tell us about your experience with VS Code so far.
Give Feedback
Remind Me Later
Ln 1, Col 1
Spaces: 4
UTF-8
LF
Plain Text

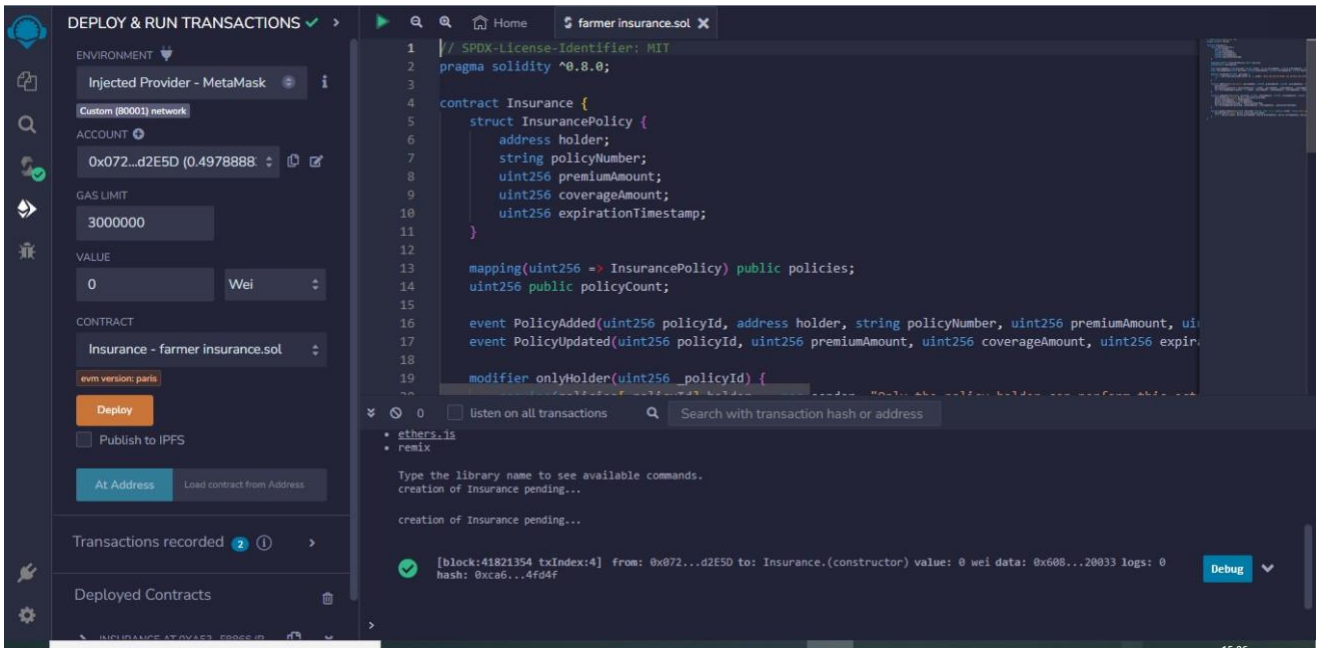




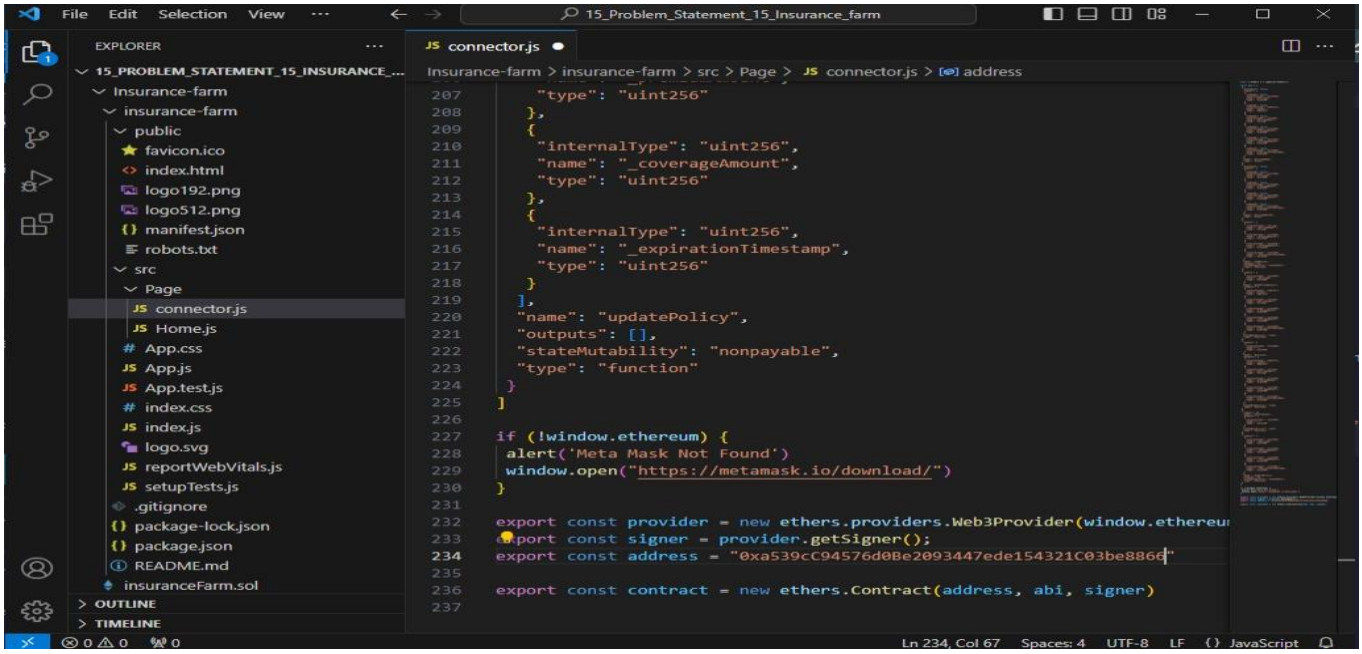
7.2 Feature 2

Remix Coding


DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
Injected Provider - MetaMask
Main (1) network
ACCOUNT
0x072...d2e5d (0 ether)
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
Insurance - farmer insurance.sol
evm version: paris
Deploy
Publish to IPFS
At Address
Load contract from Address
Transactions recorded 0
Deployed Contracts
Home
farmer insurance.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Insurance {
struct InsurancePolicy {
address holder;
string policyNumber;
uint256 premiumAmount;
uint256 coverageAmount;
uint256 expirationTimestamp;
}
mapping(uint256 => InsurancePolicy) public policies;
uint256 public policyCount;
event PolicyAdded(uint256 policyId, address holder, string policyNumber, uint256 premiumAmount, ui
event PolicyUpdated(uint256 policyId, uint256 premiumAmount, uint256 coverageAmount, uint256 expir
modifier onlyHolder(uint256 _policyId) {
require(policies[_policyId].holder == msg.sender, "Only the policy holder can perform this act
_;
}
function addPolicy(string memory _policyNumber, uint256 _premiumAmount, uint256 _coverageAmount, u
policyCount++;
policies[policyCount] = InsurancePolicy(msg.sender, policyNumber, premiumAmount, coverageAm
listen on all transactions
Search with transaction hash or address
remix
Type the library name to see available commands.
Type here to search
ENG
14:53
30-10-2023


DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
Injected Provider - MetaMask
Custom (80001) network
ACCOUNT
0x072...d2E5D (0.4978888
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
Insurance - farmer insurance.sol
evm version: paris
Deploy
Publish to IPFS
At Address
Load contract from Address
Transactions recorded 2
Deployed Contracts
Home
farmer insurance.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Insurance {
struct InsurancePolicy {
address holder;
string policyNumber;
uint256 premiumAmount;
uint256 coverageAmount;
uint256 expirationTimestamp;
}
mapping(uint256 => InsurancePolicy) public policies;
uint256 public policyCount;
event PolicyAdded(uint256 policyId, address holder, string policyNumber, uint256 premiumAmount, ui
event PolicyUpdated(uint256 policyId, uint256 premiumAmount, uint256 coverageAmount, uint256 expir
modifier onlyHolder(uint256 _policyId) {
listen on all transactions
Search with transaction hash or address
ethers.js
remix
Type the library name to see available commands.
creation of Insurance pending...
creation of Insurance pending...
[block:41821354 txIndex:4] from: 0x072...d2E5D to: Insurance.(constructor) value: 0 wei data: 0x608...20033 logs: 0
hash: 0xca6...4fd4f
Debug

9



CHAPTER-10 10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- .Transparency and Trust: Block chain ensures transparency and trust among stakeholders through its immutable ledger, enhancing confidence in the insurance process.

- Fraud Reduction: Decentralization and tamper-proof characteristics of block chain significantly reduce the risk of fraudulent activities and claims.

- Smart Contracts: Automation via smart contracts accelerates policy issuance, premium payments, and claims processing, reducing administrative overhead.

- Data Integrity: Blockchain guarantees data integrity, preventing unauthorized changes and loss, which is crucial for maintaining accurate records of insured crops and claims.

- 5Real-time Data Access: Regulators and stakeholders can access real-time data on insured crops, improving decision-making, policy enforcement, and subsidy allocation.

- Cost Efficiency: Block chain reduces paperwork and intermediaries, leading to cost savings for insurers and farmers, making insurance more affordable.

- Security and Resilience: Block chain's cryptographic security measures enhance data protection and system resilience, safeguarding sensitive information from breaches and ensuring system reliability even during network disruptions.

DISADVANTAGES :

- Technical Complexity: Integrating blockchain technology can be technically complex and may require a significant learning curve for development and maintenance.

- Scalability Challenges: Blockchain networks can face scalability issues, especially in handling a large number of transactions, which might affect the efficiency of the insurance chain.

CHAPTER-11

11. CONCLUSION

In conclusion, a Farmer Insurance Chain project utilizing blockchain technology offers numerous advantages such as transparency, reduced fraud, and automation through smart contracts. However, it also presents challenges, including technical complexity, scalability issues, and regulatory uncertainties. To succeed, careful planning, adaptation, and mitigation strategies are essential to harness the benefits of blockchain while addressing its inherent complexities and drawbacks. The success of such a project will depend on its ability to strike a balance between innovation and practicality, ensuring that it effectively serves the needs of farmers, insurers, and other stakeholders in the agricultural insurance ecosystem.

CHAPTER-12

12. FUTURE SCOPE

The future scope of a Farmer Insurance Chain project using blockchain technology holds immense potential for global expansion, technological advancements, and sustainable agriculture. Firstly, the project has the opportunity to expand its reach globally, offering farmers from diverse regions access to blockchain-based insurance, thereby enhancing agricultural sustainability on a worldwide scale. This expansion can foster greater resilience in the face of climate change and other challenges.

Secondly, through smart contract enhancements and integration with IoT and AI, the project can revolutionize insurance processes. Smart contracts can evolve to

offer more sophisticated insurance products, while the integration of IoT and AI can provide real-time data and predictive analytics, improving risk assessment and streamlining claims processing.

Lastly, the future of this project lies in its adaptability and collaboration potential. The adoption of blockchain interoperability, tokenization of agricultural assets, environmental sustainability tracking, and cross-sector collaboration can drive innovation and foster a cooperative ecosystem for farmers. By monetizing agricultural data, encouraging responsible farming practices, and facilitating cooperative risk-sharing models, blockchain technology can play a pivotal role in the agricultural insurance landscape, ultimately benefiting farmers and the global food supply chain.

CHAPTER – 12

GitHub link :

https://github.com/Shaliniviya12/blockchain.nm

Demo Link

https://drive.google.com/file/d/18pYzAwy1zVb8kkns29ZRvYKEDgthohdC/view?usp=drivesdk