Project Documentation: Decentralized Credit Scoring System (DCS)

Introduction

In decentralized finance (DeFi), overcollateralized loans are a significant hurdle, requiring users to lock up assets worth more than the loan they wish to take. This model is capital-inefficient and prevents smaller players from entering the ecosystem. To solve this, we propose a **Decentralized Credit Scoring System (DCS)** that calculates a credit score based on the on-chain lending and borrowing history of users. By leveraging transparent and verifiable data from DeFi protocols like Aave and Compound, we aim to introduce undercollateralized loans based on user creditworthiness.

Objective

The objective of this project is to enable more efficient lending by calculating and using decentralized credit scores to determine borrowing terms. Our decentralized credit scoring system will:

1. **Reduce Collateral Requirements:** Offer more favorable loan terms to users with a strong repayment history.
2. **Increase Accessibility:** Enable smaller players to access loans with less capital locked up as collateral.
3. **Leverage Existing On-chain Data:** Use data from established DeFi protocols to calculate creditworthiness.
4. **Maintain Decentralization:** Ensure the credit scoring process remains decentralized, transparent, and verifiable.

System Overview

1. Data Sources

We will use publicly available, verifiable on-chain data from DeFi protocols such as:

- **Aave:** Lending and borrowing activities.

- **Compound:** Lending, borrowing, and repayment histories.

These protocols provide transparent records of users' transaction histories, which are crucial for determining their creditworthiness.

2. Decentralized Identity (DID) Integration

Each user's credit score will be linked to their decentralized identity (DID). DIDs enable users to carry their credit history and score across multiple platforms. The decentralized identity system ensures user privacy while still allowing public verification of credit activity.

3. Credit Scoring Algorithm

The credit score will be calculated using an algorithm that evaluates the following factors:

- **Repayment History:** Timeliness and consistency of past loan repayments.
- **Loan to Collateral Ratio:** Comparison of loan amounts to collateral deposited.
- **Frequency of Borrowing:** How often the user takes loans and how long the loan period is.
- **Loan Duration:** Longer, well-managed loans boost the score.
- **Total Borrowing Volume:** A history of managing larger loans can indicate reliability.

These factors will be weighted and combined to generate a credit score that can be referenced in real-time through smart contracts.

4. Dynamic Loan Terms

Based on the credit score, loan terms will dynamically adjust:

- **Higher Scores:** Lower collateral requirements, lower interest rates, larger loan amounts.
- **Lower Scores:** Higher collateral, higher interest rates, limited loan amounts.

5. Risk Assessment Pools

Lenders can participate in risk assessment pools that group borrowers based on their credit scores. These pools dynamically adjust interest rates and collateral requirements according to the risk associated with each borrower's credit score.

Architecture

1. Blockchain Layer

- **Data Aggregation:** Smart contracts will aggregate user data from DeFi protocols like Aave and Compound.
- **Credit Score Calculation:** A smart contract will process the aggregated data and compute a user's credit score using the specified algorithm.
- **DID Integration:** Credit scores will be linked to a user's DID, enabling interoperability between different DeFi platforms.

2. Frontend Layer

- **User Interface:** The frontend will provide an intuitive dashboard where users can:
    - View their credit score.
    - Check their borrowing and repayment history.
    - Apply for loans based on their current credit score.
- **Lender Interface:** Lenders will be able to:
    - Set terms for undercollateralized loans based on the borrower's credit score.
    - Participate in different risk assessment pools.

Workflow

1. Borrower's Journey:

1. **Identity Verification:** The borrower connects their DID to the platform.
2. **Credit Score Calculation:** The platform aggregates on-chain data and computes the borrower's credit score.
3. **Loan Application:** The borrower applies for a loan based on their credit score.
4. **Terms Adjustment:** Based on the borrower's credit score, loan terms like collateral requirement and interest rate are adjusted.
5. **Loan Disbursement:** If approved, the loan is disbursed and the borrower must repay it on time to maintain or improve their score.

2. Lender's Journey:

1. **Risk Pool Selection:** The lender selects a risk pool based on the borrowers' credit scores.
2. **Loan Terms Configuration:** The lender configures loan terms for the borrowers in the pool.

3. **Loan Issuance:** Lenders can issue loans with customized terms based on the borrower's score and history.
4. **Repayment Collection:** The platform ensures timely repayments, and funds are returned to the lender, along with any accrued interest.

Smart Contract Details

1. Data Aggregator Contract

This smart contract pulls data from DeFi protocols such as Aave and Compound via oracles. It ensures the on-chain activity is accurate and up-to-date.

2. Credit Score Calculator

A smart contract that calculates and updates user credit scores based on predefined metrics:

- Timely repayments.
- Borrowing amounts.
- Loan durations.
- Loan frequency.

3. Loan Manager

This contract handles the loan application, collateral management, and loan disbursement based on the calculated credit score.

4. Risk Pool Manager

This smart contract allows lenders to group into risk pools, adjusting interest rates and collateral dynamically based on borrowers' risk profiles.

1. Blockchain & Smart Contracts

- **Ethereum/Polygon:** For deploying the smart contracts and interacting with DeFi protocols.
- **Solidity:** For writing the smart contracts that will govern the credit scoring and lending process.

2. Frontend Development

- **Next.js/React:** For building the decentralized application (dApp) frontend.
- **Tailwind CSS:** For responsive and user-friendly UI design.

3. DID Integration

- **Ethers.js:** For interacting with Ethereum, managing DID creation, and linking with the user's credit score.
- **Self-sovereign Identity (SSI) protocols:** For user authentication.

Security Considerations

- **Data Integrity:** Ensure that data pulled from external protocols is correct by using trusted oracles like Chainlink.
- **User Privacy:** Ensure that user identities and sensitive financial data are protected through encryption and decentralization.
- **Smart Contract Auditing:** Conduct regular audits to detect and resolve vulnerabilities in smart contracts.

Future Enhancements

1. **Automated Oracle Integration:**
    - Use decentralized oracles to fetch real-time data for rainfall or weather-based events to expand the credit scoring to other real-world use cases.
2. **Cross-Chain Credit Scores:**
    - Enable cross-chain compatibility, allowing users to carry their credit score between blockchains.
3. **Rewards for Good Borrowers:**
    - Introduce incentive mechanisms such as token rewards for users who consistently maintain a high credit score.

Conclusion

The Decentralized Credit Scoring System (DCS) aims to bring greater efficiency and fairness to the DeFi ecosystem by allowing undercollateralized loans based on a user's on-chain creditworthiness. By leveraging the power of decentralized data, smart contracts, and decentralized identities, we can create a more inclusive DeFi lending ecosystem.