P2P Lending System with ERC20 and NFT Repayment

1.) System Overview

The P2P Lending system is a decentralized platform that facilitates peer-to-peer lending using ERC20 tokens as currency and allows loan repayment using ERC721 tokens (NFTs). It comprises two smart contracts: ArtNFT (ERC721) and P2PLending.

2.) Smart Contract Implementation

a. ArtNFT Contract
- Implements ERC721 standard for minting and managing NFTs.
- Key features: Minting, transferring, and approval management.

b. P2PLending Contract
- Manages loan requests, guarantees, funding, and repayments.
- Implements NFT repayment functionality.

3.) Key Functionalities

a. Loan Lifecycle
- Loan Request: Borrowers create loan requests.
- Guarantee: Third parties can provide loan guarantees.
- Loan Funding: Lenders fund loan requests.
- Repayment: Borrowers repay using ERC20 tokens or propose NFT repayments.

b. NFT Repayment
- Proposal: Borrowers offer NFTs as repayment.
- Acceptance/Rejection: Lenders accept or reject NFT offers.

4.) Security Measures

a. Access Control
- Function-level checks for authorized actions.

b. State Validation
- Consistent verification of loan states.

c. Amount Validation
- Checks to prevent overpayment or invalid deductions.

d. NFT Ownership Verification
- Ensures NFT ownership and approval before transfers.

e. Reentrancy Protection
- Implements checks-effects-interactions pattern.

f. Overflow/Underflow Protection
- Utilizes Solidity 0.8.x built-in checks.

5.) Design Patterns

- Factory Pattern: For loan request creation.
- State Machine: Managing loan states.
- Pull Payment: For guaranteed withdrawals.

6.) User Interface

- Implemented using HTML and JavaScript with ethers.js.
- Features: Wallet connection, loan management, NFT repayment interface.

7.) Testing Suite

a. Structure
- Unit Tests: Individual function testing.
- Integration Tests: Complete loan lifecycle testing.
- Edge Case Tests: Extreme scenario validation.
- Fuzz Tests: Property-based testing for input validation.

b. Implementation (Using Hardhat, Mocha, and Chai)

c. Testing Desired Functionality
- Loan request creation, guarantee provision, loan funding, repayments.
- NFT repayment proposal and acceptance.
- Full loan repayment and guarantee release.

d. Testing Against Undesired Functionality
- Invalid inputs, unauthorized access attempts.
- Double spending prevention.
- State violation attempts.
- NFT ownership verification.

e. Edge Case and Fuzz Testing
- Boundary conditions, exact repayments, concurrent actions.
- Random input generation for loan requests and repayments.

f. Coverage and CI/CD

- Targets >95% code coverage.
- Integrated into CI/CD pipeline.

8.) Potential Improvements

- Implement upgradability patterns.
- Add time-lock mechanisms for critical functions.

9.) Conclusion

The P2P Lending system with NFT repayment offers an innovative approach to decentralized lending with flexible repayment options. The comprehensive implementation, security measures, and thorough testing suite ensure system reliability and robustness. However, conducting regular security assessments and updates is crucial to maintain the platform's safety and efficiency.