

Project Title:

Blockchain-Based Freelance Marketplace with Reputation Staking and Arbitration Rewards

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed Software Requirements Specification (SRS) for the development of a decentralized freelance marketplace using Cardano blockchain technology. The platform will incorporate smart contract-mediated payments, reputation-based staking, decentralized arbitration, and immutable user reputation profiles. This document outlines the functional and non-functional requirements, providing a clear understanding for developers, stakeholders, and other involved parties.

1.2 Scope

The project will facilitate transactions between clients and freelancers on a decentralized platform. It will employ smart contracts to mediate payments, a staking system to ensure accountability, and a decentralized arbitration mechanism to resolve disputes. The target users are freelancers, clients, and arbitrators, each with specific roles and interactions on the platform.

1.3 Definitions, Acronyms, and Abbreviations

- **Smart Contract:** Self-executing contract with predefined conditions stored on the blockchain.
- **Escrow:** A third-party system (in this case, a smart contract) that holds funds until project completion.
- **Staking:** The process of holding cryptocurrency as a guarantee of honest behavior.
- **Slashing:** Penalty mechanism that deducts a portion of staked tokens for misconduct or failure.
- **Arbitration:** A dispute resolution process where a neutral party (arbitrator) resolves issues between clients and freelancers.

1.4 Overview

This SRS will describe the system's functionality in sections such as system features, user requirements, system architecture, and constraints.

2. System Overview

The decentralized freelance platform will consist of:

- Smart contract-mediated escrow payments.
- Reputation staking and slashing mechanisms.
- A decentralized dispute resolution system with arbitrator rewards & accountability.
- Immutable user profiles built on blockchain reputation.

3. Functional Requirements

3.1 Smart Contract-Mediated Payments

- **Escrow Functionality:**
 - Funds from clients are held in escrow by a smart contract.

- Upon successful completion of the project, the escrowed funds are released to the freelancer.
- If a project fails or a dispute is raised, the funds remain in escrow until the issue is resolved.
- **Trigger Events:**
 - Project initiation: Contract created and funds escrowed.
 - Project completion: Client confirms and funds are released.
 - Dispute raised: Funds remain in escrow until arbitration.

3.2 Staking Mechanism

- **Reputation-Based Staking:**
 - Freelancers automatically stake 2% of their project payment.
 - The staked amount cannot be withdrawn unless the project is successfully completed.
- **Minimum Stake Requirement:**
 - Freelancers must maintain a minimum stake threshold.
- **Withdrawal of Excess Stake:**
 - Freelancers can withdraw any amount that exceeds the required minimum stake.
- **Slashing Mechanism:**
 - If a freelancer fails to deliver a project, their staked amount is slashed.
 - The slashed amount is sent to the arbitrator in case of a dispute.

3.3 Decentralized Arbitration and Rewards

- **Arbitration Trigger:**
 - A client or freelancer can trigger arbitration if they believe the terms of the contract are breached.
- **Arbitrator Selection:**
 - Arbitrators are selected from a decentralized pool.
 - The decision process is transparent.
- **Arbitrator Rewards:**
 - If **one party** is fully wrong, the arbitrator receives their slashed stake.
 - If **both parties** are fully wrong, the arbitrator receives the slashed amounts from both.
 - If **both parties** are partially wrong, 50% of each party's stake is slashed and awarded to the arbitrator.

3.3.1 Arbitrator Selection and Staking

- **Minimum Stake Requirement for Arbitrators:**
 - Arbitrators must have a stake amount equal to or greater than the reward they are set to receive. This ensures that arbitrators have a vested interest in providing fair and unbiased decisions.
- **Stake Locking Mechanism:**
 - The arbitrator's staked amount will be locked for 14 days after the resolution of a dispute. This serves as a security measure in case the arbitrator is later found to have acted with bias or impropriety.
- **Dispute Against Arbitrators:**
 - If a freelancer or client believes the arbitrator was not neutral or acted unfairly, they can raise a dispute against the arbitrator.
 - This dispute will be escalated to a **Level 2 Arbitrator**, who is verified by the system and holds a high reputation on the platform.

3.3.2 Level 2 Arbitrator Resolution

- **High Reputation and Verification:**
 - Level 2 Arbitrators are selected based on their reputation and experience in the system. They are thoroughly vetted to ensure integrity in their decision-making process.
- **Penalty for Non-Neutral Arbitrators:**
 - If the original arbitrator is found to have acted unfairly or with bias, their staked amount will be forfeited.
 - **Staked Amount Redistribution:**
 - The staked amount of the original arbitrator will be awarded to the Level 2 Arbitrator as a reward for their judgment.
 - The original reward amount (meant for the arbitrator) will go to the wronged party (either the freelancer or the client) as compensation.

3.4 User Profiles and Blockchain-Based Reputation

- **Immutable Reputation:**
 - Freelancers and clients have profiles built on blockchain-based transactions, ensuring transparency and resistance to manipulation.
 - Reputation scores are updated based on completed projects and disputes.
- **User Reviews:**
 - Clients leave reviews and ratings for freelancers after each completed project, contributing to their overall reputation score.
- **Profile View:**

- Reputation and past projects are publicly viewable, enabling clients and freelancers to assess each other's history before entering into agreements.

3.5 Slashing History and Transparency

- **Slashing Record:**
 - The system will track and display the number of times a user's reward or stake has been slashed due to non-delivery or arbitration outcomes.
 - This information will be included in the user's profile, contributing to their overall reputation score.
- **Transparency for Clients & Freelancer:**
 - Clients/Freelancer can view the slashing history of freelancers/client before entering into contracts, ensuring transparency regarding past performance and reliability.

4. Non-Functional Requirements

4.1 Security

- **Smart Contract Security:**
 - Contracts must be audited and tested to prevent vulnerabilities like re-entrancy attacks or funds freezing.
- **Data Integrity:**
 - Immutable blockchain records ensure that transaction history, user profiles, and reputations cannot be tampered with.

4.2 Performance

- **Transaction Speed:**
 - Smart contract transactions (escrow and staking) should be confirmed within a reasonable timeframe, balancing speed with network security.

4.3 Scalability

- The system must support a large number of simultaneous contracts, payments, and arbitrations without performance degradation.

4.4 Usability

- **User-Friendly Interface:**
 - The platform should offer an intuitive interface for freelancers, clients, and arbitrators to interact seamlessly with smart contracts.

5. System Architecture

5.1 Smart Contracts

- Each transaction, escrow, and staking event is handled via smart contracts on the Cardano blockchain.

5.2 Blockchain

- **Cardano:** The project will be built on the Cardano blockchain, leveraging its secure, scalable, and energy-efficient Proof of Stake (PoS) consensus mechanism.

5.3 Dispute Resolution Engine

- The decentralized arbitration system is connected to the smart contracts, allowing for transparent, on-chain decision-making and fund redistribution.

6. Assumptions and Constraints

- **Assumptions:**
 - Users are familiar with cryptocurrency and blockchain transactions.
 - Users will manage their own wallets and private keys.
- **Constraints:**
 - Transactions are dependent on Cardano blockchain network speed and fee structure.
 - All funds and reputation data are tied to the blockchain, meaning any mistakes (e.g., lost private keys) cannot be rectified by the platform.

7. Future Enhancements

- **Arbitration Appeals:**
 - Potential introduction of an appeals process for users who dispute the arbitrator's decision.
- **Automated Dispute Resolution:**
 - Implement AI-assisted arbitration for smaller disputes to speed up resolution times.