

# Functional Requirements Document (FRD)

**Project Title:** Stake Chain

**Version:** 1.0

**Document Status:** Draft

**Prepared By:** Endi Troqe

Computer Science Student

# TABLE OF CONTENTS

<b>2</b>	<b>GENERAL.....</b>	<b>3</b>
2.1	PROJECT DESCRIPTION.....	3
2.1.1	Background.....	3
2.1.2	Purpose.....	3
2.1.3	Assumptions and Constraints.....	3
2.1.4	Interfaces to External Systems.....	4
2.2	POINTS OF CONTACT.....	4
2.3	DOCUMENT REFERENCES.....	4
<b>3</b>	<b>FUNCTIONAL REQUIREMENTS.....</b>	<b>4</b>
3.1	DATA REQUIREMENTS.....	4
3.2	FUNCTIONAL PROCESS REQUIREMENTS.....	4
3.2.1	Setup.....	4
3.2.2	During game.....	5
3.2.3	Extra buttons.....	5
<b>4</b>	<b>Operational requirements.....</b>	<b>5</b>
4.1	SECURITY.....	5
4.2	AUDIT TRAIL.....	5
4.3	DATA CURRENCY.....	6
4.4	RELIABILITY.....	5
4.5	PERFORMANCE.....	6
4.6	CAPACITY.....	6
4.7	DATA RETENTION.....	6
<b>5</b>	<b>REQUIREMENTS TRACEABILITY MATRIX.....</b>	<b>6</b>
<b>6</b>	<b>Glossary.....</b>	<b>6</b>

# 1 GENERAL

## 1.1 Project Description

This project involves developing a blockchain-based group betting platform that allows users to create, join, and manage custom bets within groups. Each group is managed by a **group leader**, who has complete control over group operations, including game resets, coin allocations, and bet creation.

### 1.1.1 Background

This platform is designed to offer a fun and engaging way for users to gamify group interactions by making predictions and wagers on custom scenarios. It's suitable for various contexts, including academic, personal, and sports-related bets.

### 1.1.2 Purpose

The primary objective is to provide a transparent, secure, and customizable betting experience. By leveraging blockchain, the platform ensures fairness, immutability, and secure transactions while promoting social interaction and competition within groups.

### 1.1.3 Assumptions and Constraints

- Users have access to a blockchain wallet for managing coins.
- Blockchain ensures transparency and immutability of all transactions.
- The bookie is trusted to manage the group fairly.

Constraints:

- The platform must operate within local legal frameworks for betting.
- Transaction fees on the blockchain should be minimized.

Preference:

- User friendly button setup on the remote
- Visually appealing buzzer design

#### 1.1.4 1.1.4 Interfaces to External Systems

- **Blockchain Network:** For smart contract execution and coin transactions.
- **Frontend Systems:** Modern frameworks like React.js for a user-friendly interface.
- **Off-Chain Database:** For storing non-critical metadata (e.g., leaderboard statistics).

## 1.2 Points of Contact

Student – Endi Troqe

Faculty advisor – Dr. Kancharla

## 1.3 Document References

- Blockchain Smart Contract Standards (e.g., ERC-20 for coins).

## 1.4 Data Requirements

*Data Stored On-Chain:*

- Bet details (description, type, threshold, etc.).
- User stakes and selections.
- Game reset events and timestamps.

*Data Stored Off-Chain:*

- User profiles (excluding wallet details).
- Leaderboard statistics.
- Metadata for bet descriptions.

# 2 FUNCTIONAL PROCESS REQUIREMENTS

### 2.1.1 Setup

- Bet Creation
  - **Input:** The group leader defines the bet type (Yes/No or Over/Under), description, rules, and minimum stake.

- **Output:** Smart contract handles the bet, locks stakes, and processes payouts after the result is submitted.
- Bet Participation
  - **Input:** Users stake coins and select an option.
  - **Output:** Smart contract records stakes and choices, locking coins until the bet is resolved.
- Coin Management
  - **Input:** Bookie assigns initial coins during group creation.
  - **Input:** Bookie can manually increase a user's coin balance as needed.
  - **Output:** Coin balances are updated securely on the blockchain.
- Leaderboards
  - **Input:** Smart contract tracks users' coin holdings and updates rankings.
  - **Output:** Group leaderboards dynamically display rankings.
- **Game Reset**
  - **Input:** Bookie initiates a reset to restart the game.
  - **Output:** Coins, bets, and leaderboards are reset to the starting state.

## 2.2 Security

- Blockchain ensures all transactions are immutable and transparent.
- Access to coins and bets is tied to secure blockchain wallets.

## 2.3 Audit trail

- All bet details, stakes, and results are recorded on-chain.
- Game Resets

## 2.4 Reliability and Performance

- Smart contracts execute transactions within 1 second.
- Maintains performance 99.9% of time

## 2.5 Capacity and Data Retention

- Supports up to 10,000 concurrent users.
- Bet and leaderboard data are retained until a game reset is triggered.

### 3 REQUIREMENTS TRACEABILITY MATRIX

#	Requirement Description	Requirement Reference in FRD	Verification Method	Requirement Reference in Test Plan
1	Bet creation and participation process	Section 4.2 Setup	Validate bets are created and joined.	Test 1: Bet creation.
2	Coin management by Group leader	Section 4.2 Timer Button	Test coin allocation functionality.	Test 2: Coin allocations.
3	Group-based leaderboards	Section 4.2 Start Button	Verify rankings update dynamically.	Test 3: Leaderboard updates.
4	Game reset functionality.	Section 4.2 Stop Button	Confirm successful reset process.	Test 4: Reset functionality.

### 4 GLOSSARY

- **Group Leader:** The group leader responsible for managing bets, coin allocations, and resets.
- **Stake:** Coins wagered by users in a bet.
- **Smart Contract:** Blockchain-based code that automates betting, payouts, and resets.
- **Reset:** A process where the bookie restarts the group, resetting coins, leaderboards, and bets.